To: T10 Technical Committee
From: Rob Elliott, HP (elliott@hp.com)
Date: 2 November 2005
Subject: 05-381r0 SAS-2 Multiplexing

Revision history

Revision 0 (2 November 2005) First revision

Related documents

sas2r00 - Serial Attached SCSI - 2 revision 00

Overview

When Serial Attached SCSI was first conceived, it included the concept of time division multiplexing a physical link into two logical links when a 3 Gbps HBA is talking to multiple (SATA) 1.5 Gbps disk drives. This feature was removed before submittal to T10 to reduce protocol complexity. If a 3 Gbps HBA talks to a 1.5 Gbps disk drive, rate matching is used - ALIGN/NOTIFYs are inserted every other dword and half the bandwidth on the 3 Gbps is wasted.

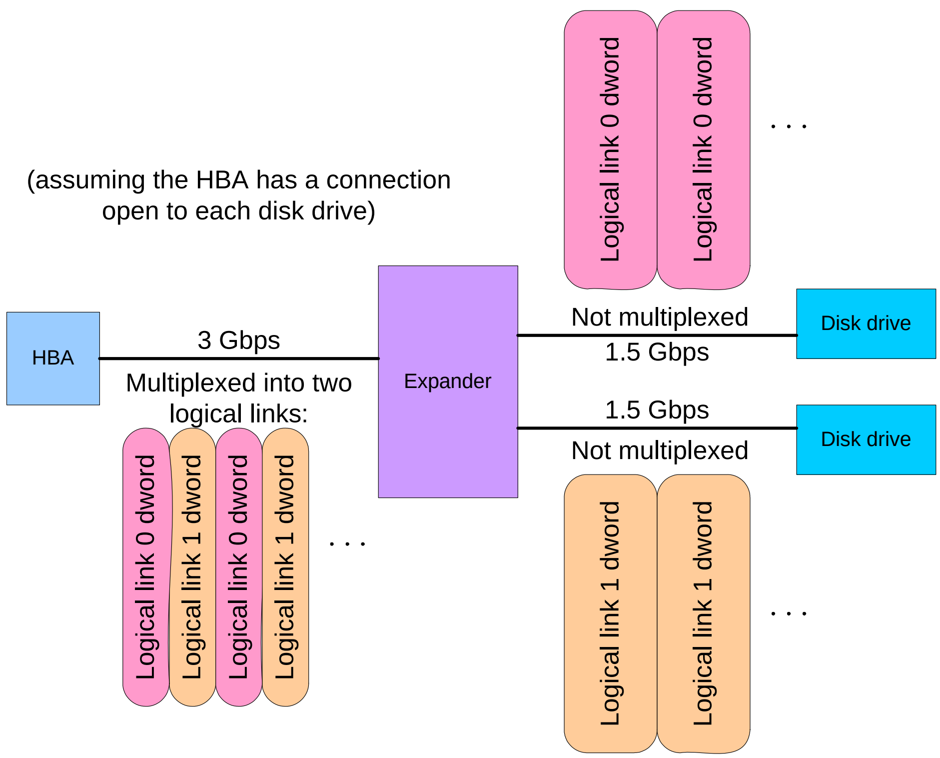


Figure 1 — Multiplexing overview

Multiplexing reclaims that bandwidth by transmitting dwords for another connection rather than ALIGN/NOTIFYs.

Key points of the proposal:

a) Multiplexing is optional and can be done by any phy faster than 1.5 Gbps
b) Multiplex can be done by both end devices (both initiators and targets) and expander devices
c) Static multiplexing (not dynamic)
   A) One-way multiplexing (means no multiplexing)
   B) Two-way multiplexing (e.g. a 3 Gbps physical link into two 1.5 Gbps logical links, or a 6 Gbps physical link into two 3 Gbps logical links)
   C) Four-way multiplexing (e.g. a 6 Gbps physical link into four 1.5 Gbps logical links)
   D) No nesting (e.g. no 6 Gbps physical link to one 3 Gbps and two 1.5 Gbps logical links)
d) IDENTIFY address frame modifications

   A) Indicate support for multiplexing (none, 2-way, 4-way)
   B) Indicate true bandwidth needs (for targets)
e) Negotiate to turn on multiplexing after the link reset sequence
   A) Define new MUX(0)/(1)/(2)/(3) primitives
   B) Transmit MUX and receive MUX to begin multiplexing
   C) Only transmit MUX if both IDENTIFY address frames agreed that multiplexing is supported
   D) Negotiate to the highest common level of multiplexing supported by the two phys (e.g. favor 4-way over 2-way, but 1-way wins over all)
   E) Rotate between MUX (0)/(1)/(2)/(3) to reduce EMI while negotiating
   F) MUXing happens outside elasticity buffers, so no ALIGN/NOTIFY during MUX exchange
f) Rerun link reset sequence if multiplexing level needs to be changed
g) Rerun link reset sequence on loss of dword synchronization (since which dword belongs to which logical link is uncertain)
h) No modifications to the SL_CC state machine; it just works on logical phys rather than physical phys
i) Clock skew management
   A) Multiplex after clock skew management ALIGN/NOTIFY insertion
   B) Demultiplex before the elasticity buffers
   C) Multiplex every other dword, not every other non-ALIGN/NOTIFY
   D) ALIGN/NOTIFY frequency within each logical link must equal that of a physical link at the same rate
j) SMP functions
   A) In PHY CONTROL, provide field to specify how many logical links a phy should request in the IDENTIFY address frame
   B) In DISCOVER, report current multiplexing status (enabled/disabled), outgoing IDENTIFY content, incoming IDENTIFY content
k) Discover process algorithm
   A) decides whether or not to request multiplexing
   B) high-speed targets need to be able to make high-speed connections; others get the leftovers

Additionally, non-multiplexing specific changes are included to support 6 Gbps throughout the protocol layers.

Alternatives

One alternative is to let the expander handle everything; e.g., let HBAs speak to expanders at 6 Gbps, expanders speak to drives at 3 Gbps, and have the expander terminate the connections on each side and store-and-forward multiple frames accumulated during the connection. This is very complicated for the expander and may require protocol changes to optimize performance. The SAS connection-based fabric is not well suited for a packet-switched approach - telephone networks do this type of conversion, but voice connections have low bandwidth requirements.

Another alternative is a dynamic rather than static multiplexing scheme. This would let connections with different connection rates share the same physical link without requiring redoing the link reset sequence. After an OPEN (3 Gbps) is sent, every other dword is available to carry another connection, not just rate matching ALIGN/NOTIFYs. Problems that would have to be solved with a dynamic scheme include:

a) maintain proper ALIGN/NOTIFY insertion rates
b) keep the ability to do rate matching within a 3 Gbps connection
c) avoid starvation of 6 Gbps connection requests by 3 Gbps connections that keep slipping in

Suggested changes to SAS-2

Changes to the model clause

Add logical phy and logical link terms.

3.1 Definitions

3.1.1 attached SAS address: The SAS address (see 3.1.165) of the attached phy (e.g., received by a physical phy in the incoming IDENTIFY address frame during the initialization sequence (see 4.1.2)), or the SAS address of the STP target port in an STP/SATA bridge (see 4.6.2).

3.1.2 connection rate: The effective rate of dwords through the pathway between a SAS initiator phy and a SAS target phy, established through the connection request.

3.1.3 expander phy: A phy in an expander device that interfaces to a service delivery subsystem.

3.1.4 logical link: A physical link or a multiplexed portion of a physical link. See 4.x.

3.1.5 logical phy: A phy or a multiplexed portion of a phy. See 4.x.

3.1.6 multiplexing: A Dividing a physical link into multiple logical links. See 4.xx.

3.1.7 partial pathway: The set of ~~physical~~logical links participating in a connection request that have not yet conveyed a connection response. See 4.1.9.

3.1.8 pathway: A set of ~~physical~~logical links between a SAS initiator phy and a SAS target phy being used by a connection. See 4.1.9.

3.1.9 phy: A object in a device that is used to interface to other devices (e.g., an expander phy (see 3.1.3) or a SAS phy (see 3.1.14)). See 4.1.2.

3.1.10 physical link: Two differential signal pairs, one pair in each direction, that connect two physical phys. See 4.1.2.

3.1.11 physical phy: A phy (see 3.1.9) that contains a transceiver (see 3.1.241) and electrically interfaces to a physical link to communicate with another physical phy. See 4.1.2.

3.1.12 potential pathway: A set of ~~physical~~logical links between a SAS initiator phy and a SAS target phy. See 4.1.9.

3.1.13 rate: Data transfer rate of a physical link (e.g., 1,5 Gbps~~ or~~, 3~~,0~~ Gbps, or 6 Gbps).

3.1.14 SAS phy: A phy in a SAS device that interfaces to a service delivery subsystem.

3.1.15 unit interval (UI): The normalized, dimensionless, nominal duration of a signal transmission bit (e.g., 666,$\overline{6}$ ps at 1,5 Gbps~~ and~~, 333,$\overline{3}$ ps at 3~~,0~~ Gbps, and 166,6 ps at 6 Gbps). Unit interval is a measure of time that has been normalized such that 1 UI is equal to 1/baud seconds.

3.1.16 virtual phy: A phy (see 3.1.9) that interfaces with a vendor-specific interface to another virtual phy inside the same device. See 4.1.2.

3.2 Symbols and abbreviations

See 2.1 for abbreviations of standards bodies (e.g., ISO). Units and abbreviations used in this standard:

Abbreviation	Meaning
AA	ATA application layer (see 10.3)
A.C.	alternating current
ACK	acknowledge primitive (see 7.2.6.1)
AIP	arbitration in progress primitive (see 7.2.5.1)
ATA	AT attachment (see 3.1.10)
ATAPI	AT attachment packet interface
ATA/ATAPI-7	AT Attachment with Packet Interface - 7 standard (see 2.3)
AWG	American wire gauge
AWT	arbitration wait time

Abbreviation	Meaning
BCH	Bose, Chaudhuri and Hocquenghem code (see 4.2.3)
BER	bit error ratio (see 3.1.16)
BIST	built in self test
G1	generation 1 physical link rate (1,5 Gbps)
G2	generation 2 physical link rate (3~~,0~~ Gbps)
G3	generation 3 physical link rate (6 Gbps)
G~~3~~4	generation ~~3~~4 physical link rate (defined in a future version of this standard)
Gbps	gigabits per second (10^9 bits per second)
Gen1i	SATA generation 1 physical link rate (1,5 Gbps)(see SATAII-PHY)
Gen1x	SATA generation 1 physical link rate (1,5 Gbps), extended length (see SATAII-PHY)
Gen2i	SATA generation 2 physical link rate (3,0 Gbps)(see SATAII-PHY)
Gen2x	SATA generation 2 physical link rate (3,0 Gbps), extended length (see SATAII-PHY)

Changes to the model clause

Add the concept of logical phys and logical links to the model.

4 General

4.1.2 Physical links and phys

A physical link is a set of four wires used as two differential signal pairs. One differential signal transmits in one direction while the other differential signal transmits in the opposite direction. Data may be transmitted in both directions simultaneously.

A physical phy contains a transceiver which electrically interfaces to a physical link, which attaches to another physical phy. A virtual phy contains a vendor-specific interface to another virtual phy.

Phys are contained in ports (see 4.1.3). Phys interface to the service delivery subsystem (see 4.1.6).

Figure 2 shows two phys attached with a physical link.

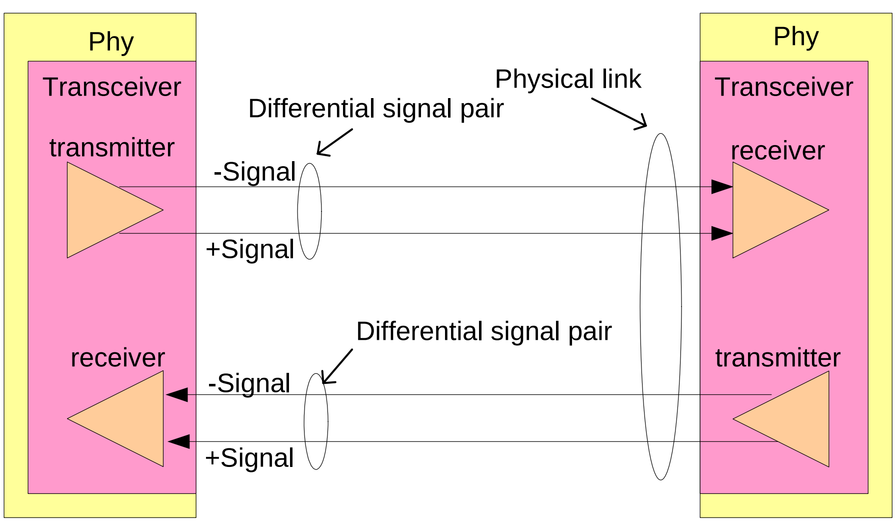


Figure 2 — Physical links and phys

An attached phy is the phy to which a phy is attached over a physical link.

A device (i.e., a SAS device (see 4.1.4) or expander device (see 4.1.5)) contains one or more phys.

Each phy has:

a) a SAS address (see 4.2.2), inherited from the SAS port (see 4.1.3) or expander device;
b) a phy identifier (see 4.2.7) which is unique within the device;
c) optionally, support for being an SSP initiator phy;
d) optionally, support for being an STP initiator phy;
e) optionally, support for being an SMP initiator phy;
f) optionally, support for being an SSP target phy;
g) optionally, support for being an STP target phy; and
h) optionally, support for being an SMP target phy.

During the identification sequence (see 7.9), a phy:

a) transmits an IDENTIFY address frame including the device type (i.e., end device, edge expander device, or fanout expander device) of the device containing the phy, the SAS address of the SAS port or expander device containing the phy, phy identifier, SSP initiator phy capability, STP initiator phy capability, SMP initiator phy capability, SSP target phy capability, STP target phy capability, and SMP target phy capability.
b) receives an IDENTIFY address frame containing the same set of information from the attached phy, including the attached device type, attached SAS address, attached phy identifier, attached SSP initiator phy capability, attached STP initiator phy capability, attached SMP initiator phy capability, attached SSP target phy capability, attached STP target phy capability, and attached SMP target phy capability.

The transceiver follows the electrical specifications defined in 5.3. Phys transmit and receive bits at physical link rates defined in 5.3. The physical link rates supported by a phy are specified or indicated by the NEGOTIATED PHYSICAL LINK RATE field, HARDWARE MINIMUM PHYSICAL LINK RATE field, the HARDWARE MAXIMUM PHYSICAL LINK RATE field, the PROGRAMMED MINIMUM PHYSICAL LINK RATE field, and the PROGRAMMED MAXIMUM PHYSICAL LINK RATE field in the SMP DISCOVER function (see 10.4.3.5), SMP PHY CONTROL function (see 10.4.3.10), and Phy Control and Discover subpage (see 10.2.7.2.3). The bits are part of dwords (see 6.2.1), each of which has been encoded using 8b10b coding into four 10-bit characters (see 6.2).

A phy may be used as one, two, or four logical phys based on multiplexing (see 7.xx).

Figure 3 defines the phy classes, showing the relationships between the following classes:

a) phy;
b) SAS phy;
c) expander phy;
d) SAS initiator phy;
e) SAS target phy;
f) SSP phy;
g) STP phy; and
h) SMP phy.

SATA phys are also referenced in this standard but are defined by SATA (see ATA/ATAPI-7 V3).

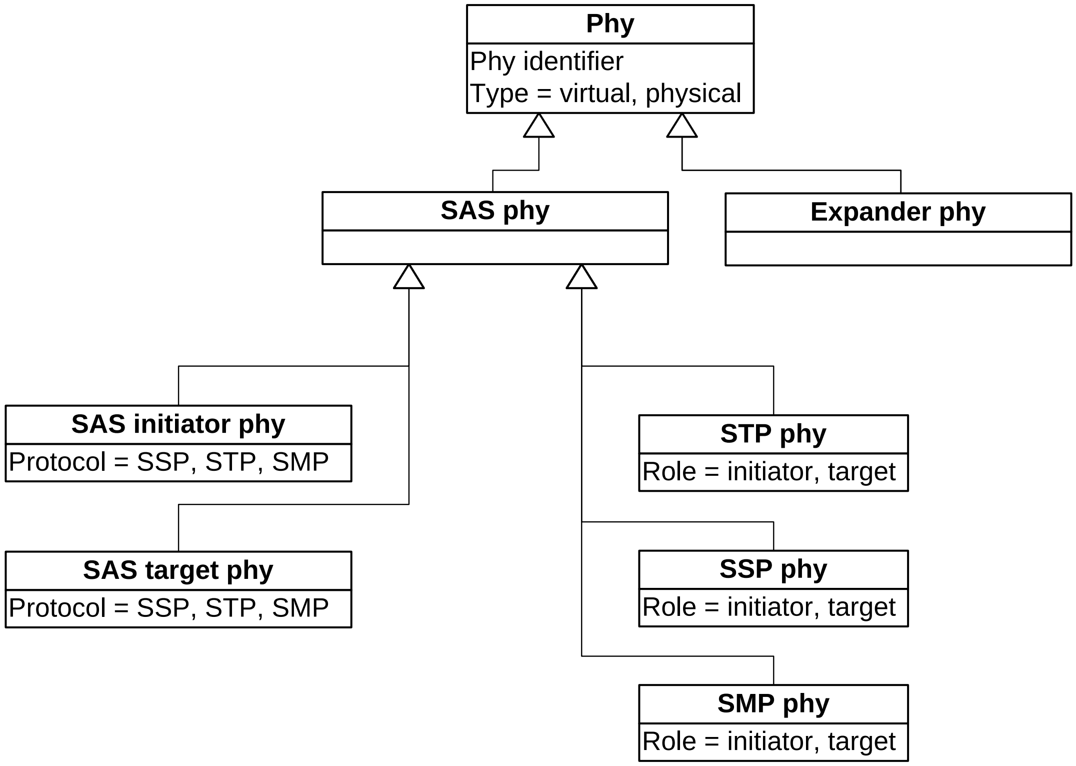


Figure 3 — Phy class diagram

Editor's Note 1: Logical phys need to be included in the UML model, perhaps just as another attribute or a new value in the Type attribute.

Figure 4 shows the objects instantiated from the phy classes, including:

a) from the SAS phy class:
   A) SSP initiator phy;
   B) SSP target phy,
   C) virtual SSP initiator phy;
   D) virtual SSP target phy;
   E) STP initiator phy;
   F) STP target phy;
   G) virtual STP initiator phy;
   H) virtual STP target phy;
   I) SMP initiator phy;
   J) SMP target phy;
   K) virtual SMP initiator phy; and
   L) virtual SMP target phy;

   and

b) from the expander phy class:
   A) expander phy; and
   B) virtual expander phy.

A phy is represented by one of these objects during each connection. A phy may be represented by different phy objects in different connections.

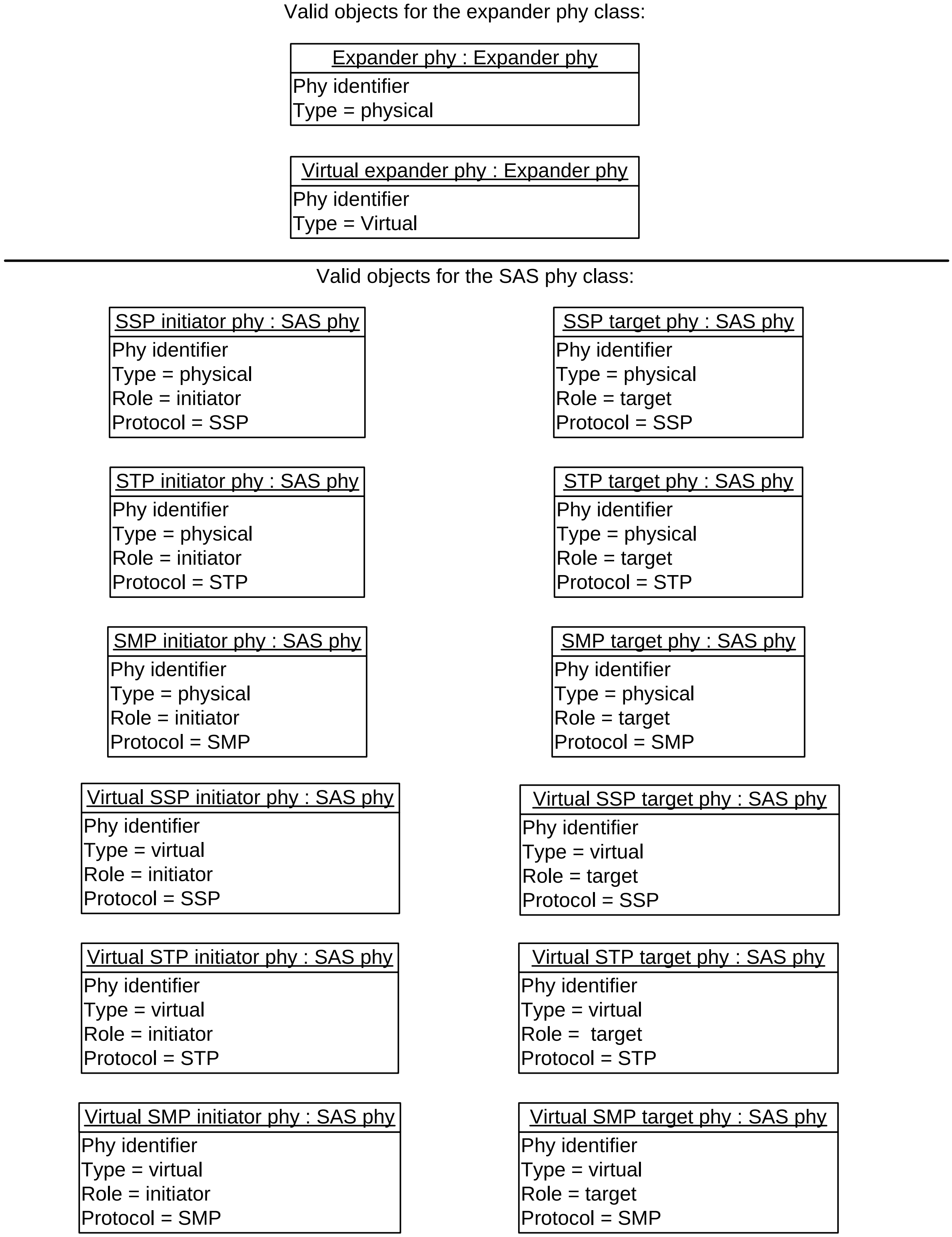


Figure 4 — Phy object diagram

4.x Logical links

A physical link with a physical link rate greater than 1,5 Gbps may be multiplexed into two or four logical links as defined in table 1.

Table 1 — Logical links

Physical link rate	Logical links
6 Gbps	One 1,5 Gbps logical link
	Two 3 Gbps logical links
	Four 1,5 Gbps logical links
3 Gbps	One 3 Gbps logical links
	Two 1,5 Gbps logical links
1,5 Gbps	One 1,5 Gbps logical link

Logical links are negotiated using MUX primitives (see 7.xx).

4.3.1 State machine overview

Figure 5 shows the state machines for SAS devices, their relationships to each other and to the SAS device, SAS port, and SAS phy classes.

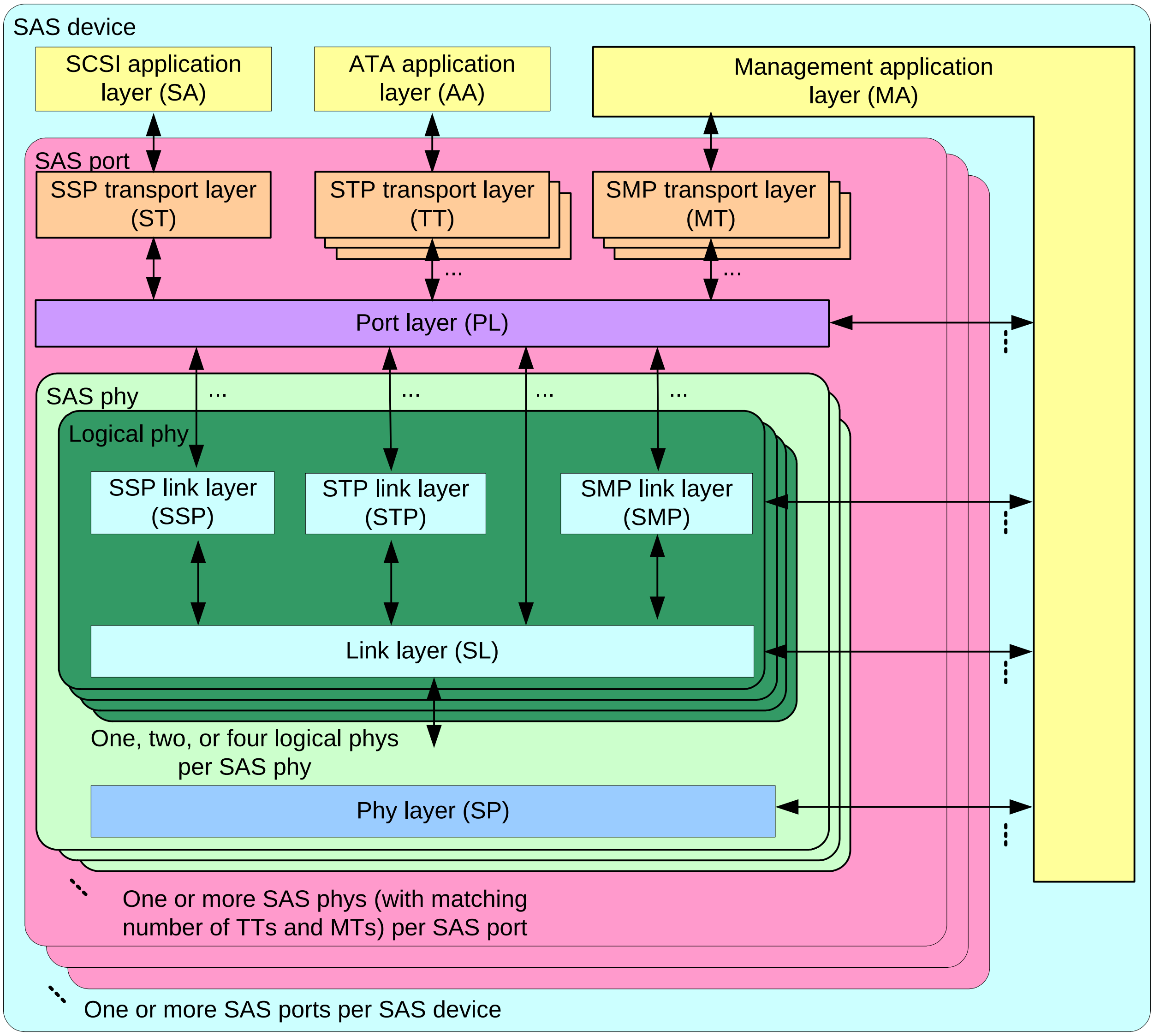


Figure 5 — State machines for SAS devices [updated to include logical phys]

Figure 6 shows the state machines for expander devices, their relationships to each other and to the expander device, expander port, and expander phy classes. Expander function state machines are not defined in this standard, but the interface to the expander function is defined in 4.6.6.

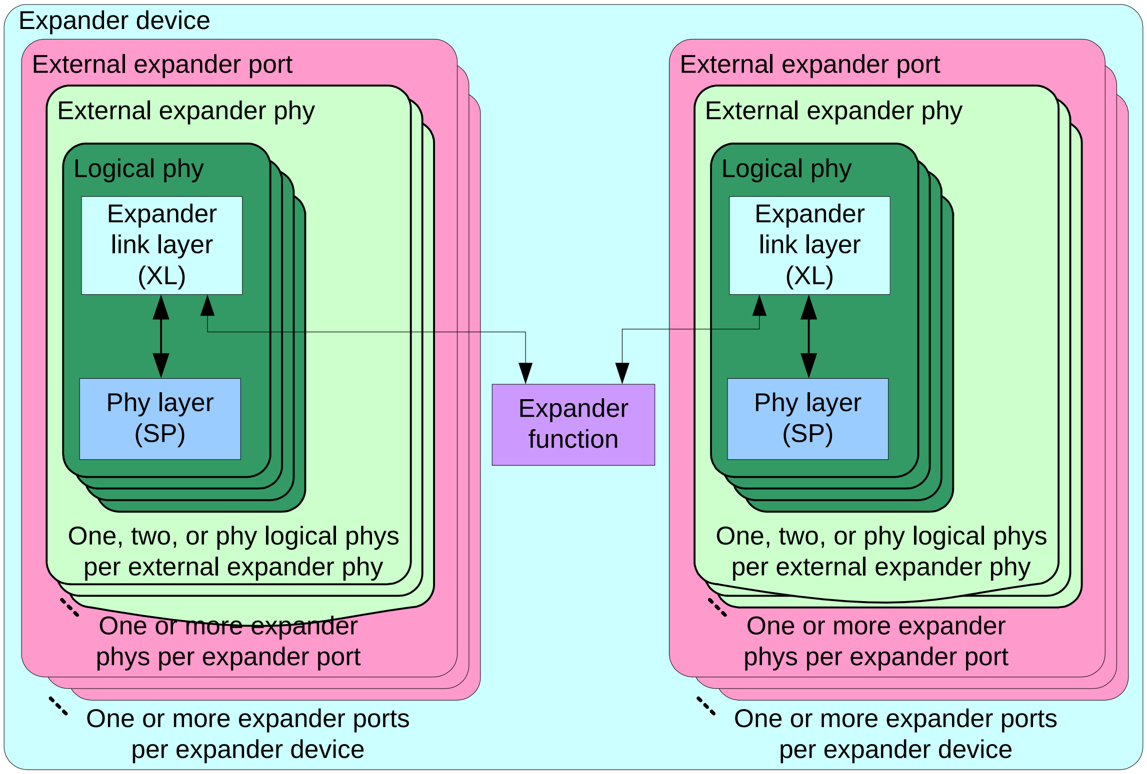


Figure 6 — State machines for expander devices [updated to include logical phys]

Annex K contains a list of messages between state machines.

4.3.2 Transmit data path

Figure 7 shows the transmit data path in a SAS phy.

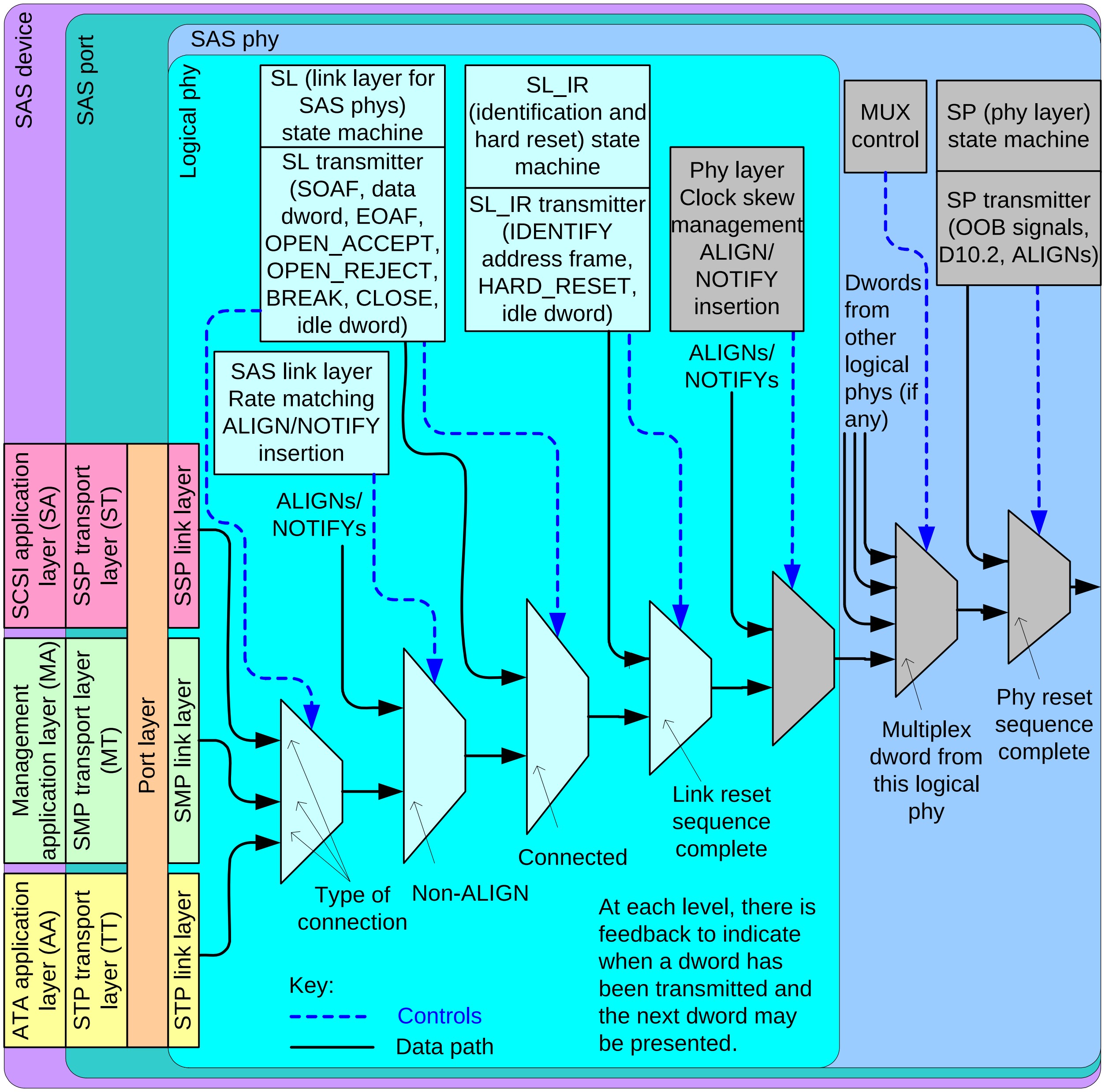


Figure 7 — Transmit data path in a SAS phy [updated to include logical phy]

...

Figure 8 shows the transmit data path in an expander phy.

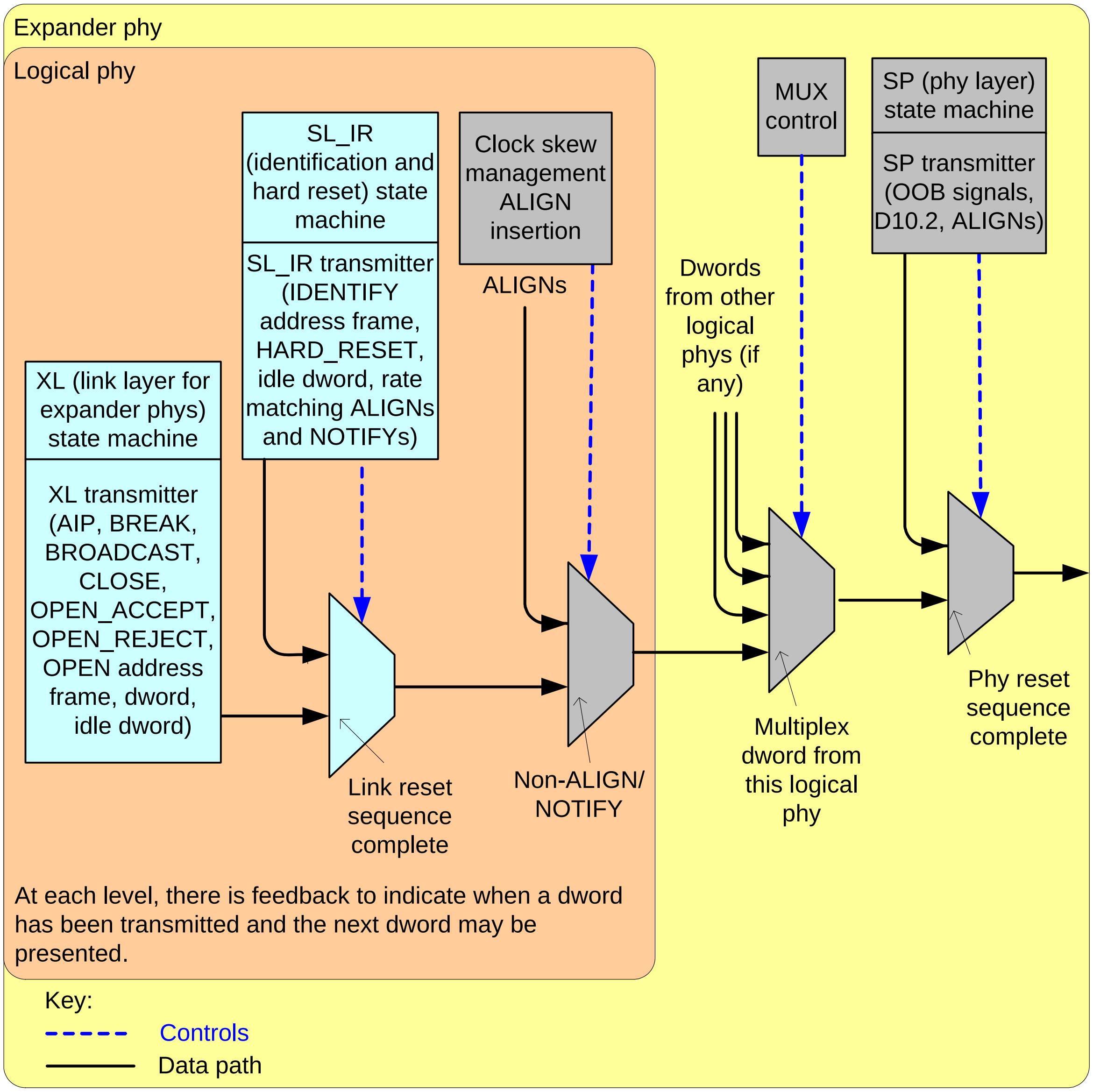


Figure 8 — Transmit data path and state machines in an expander phy [updated to include logical phy]

4.3.3 Receive data path

The SP_DWS receiver (see 6.9.2) establishes dword synchronization and sends dwords to the SP_DWS state machine (see 6.9) and to the link layer state machine receivers.

Figure 9 shows the receive data path in a SAS phy.

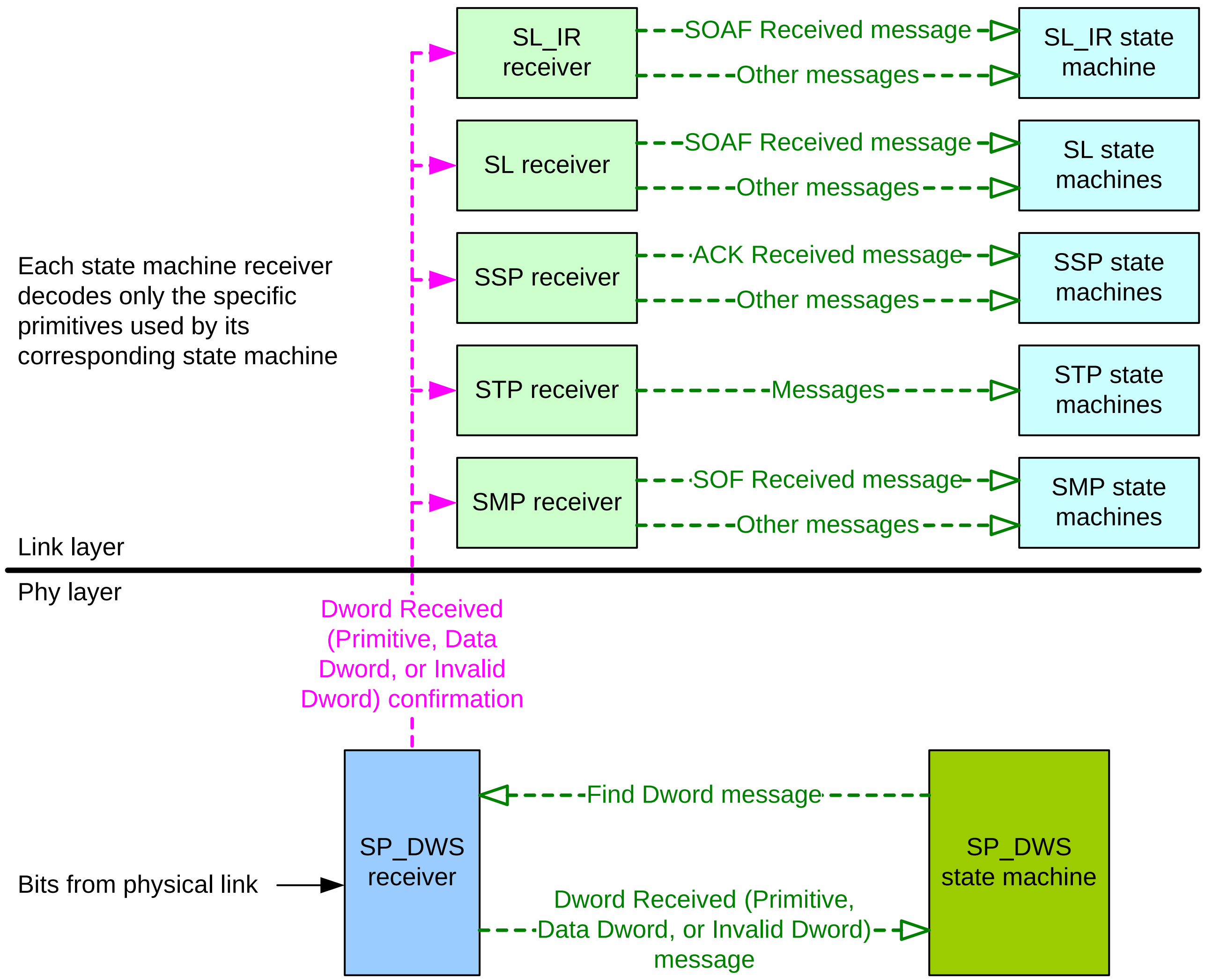


Figure 9 — Receive data path in a SAS phy [no change]

Figure 10 shows the receive data path in an expander phy.

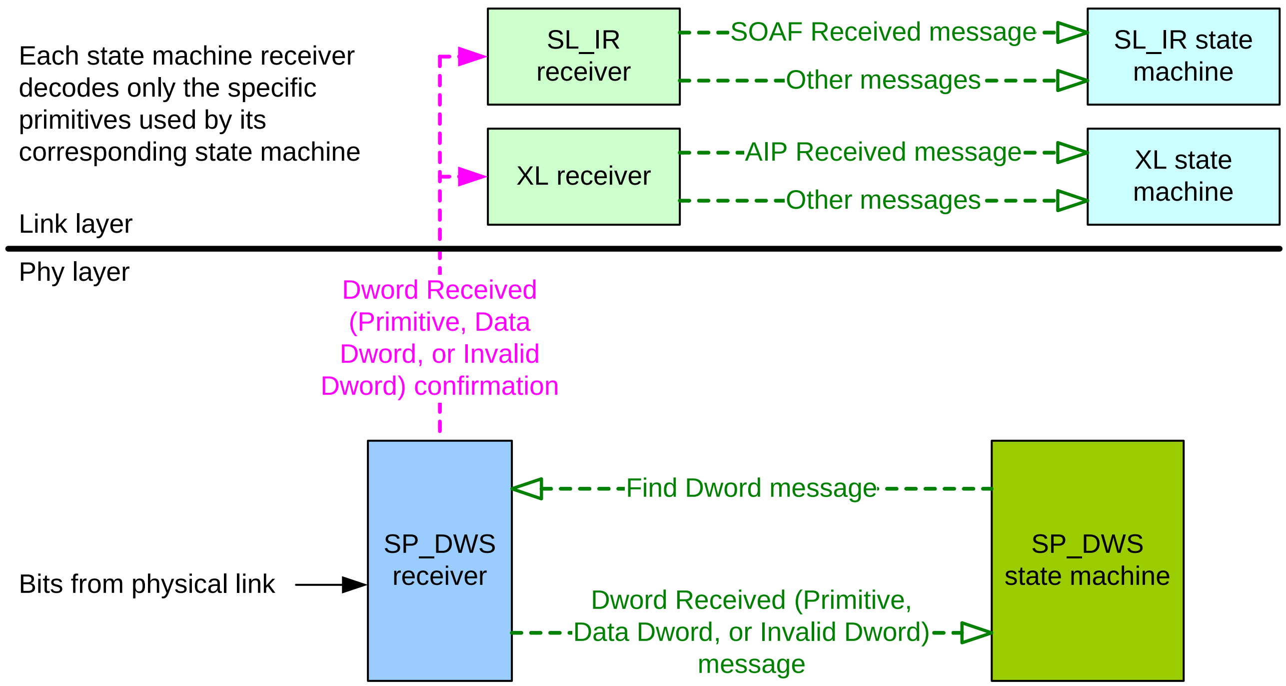


Figure 10 — Receive data path in an expander phy [no change]

4.1.9 Pathways

A potential pathway is a set of ~~physical~~logical links between a SAS initiator phy and a SAS target phy. When a SAS initiator phy is directly attached to a SAS target phy with a non-multiplexed physical link, there is one potential pathway. When the physical link is multiplexed or there are expander devices between a SAS initiator phy and a SAS target phy, it is possible that there is more than one potential pathway, each consisting of a set of ~~physical~~logical links between the SAS initiator phy and the SAS target phy. The physical links may or may not be using the same physical link rate.

A pathway is a set of ~~physical~~logical links between a SAS initiator phy and a SAS target phy being used by a connection (see).

Figure 11 shows examples of potential pathways.

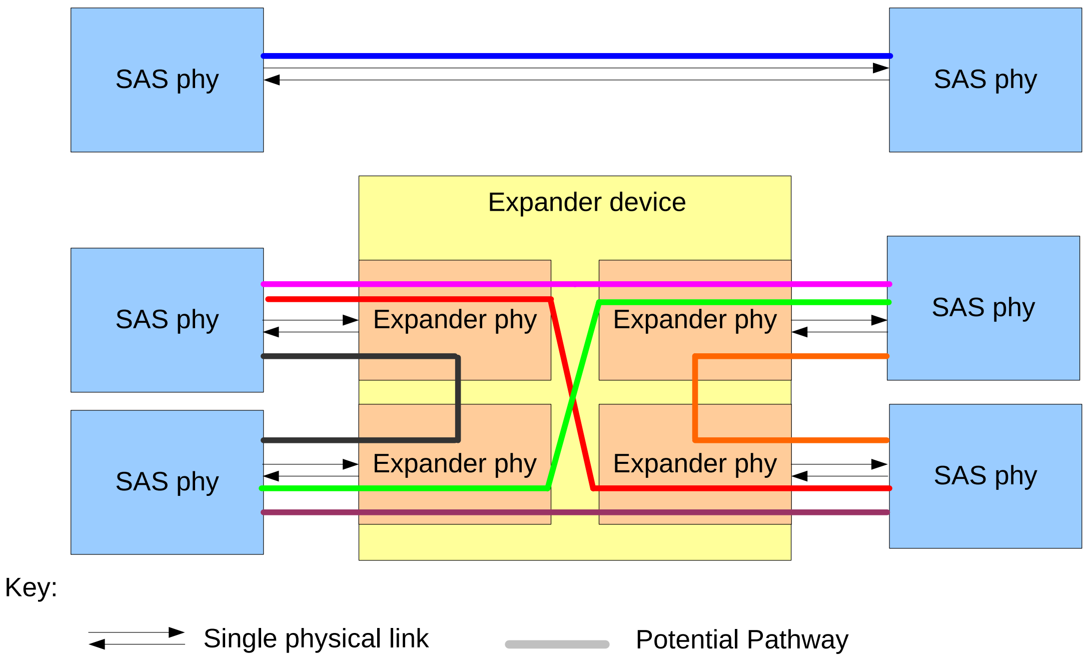


Figure 11 — Potential pathways

A partial pathway is the set of ~~physical~~logical links participating in a connection request that have not yet conveyed a connection response (see 7.12).

A partial pathway is blocked when path resources it requires are held by another partial pathway (see 7.12).

4.1.10 Connections

A connection is a temporary association between a SAS initiator port and a SAS target port. During a connection all dwords from the SAS initiator port are forwarded to the SAS target port, and all dwords from the SAS target port are forwarded to the SAS initiator port.

A connection is pending when an OPEN address frame has been delivered along a completed pathway to the destination phy but the destination phy has not yet responded to the connection request. A connection is established when an OPEN_ACCEPT is received by the source phy.

A connection enables communication for one protocol: SSP, STP, or SMP. For SSP and STP, connections may be opened and closed multiple times during the processing of a command (see 7.12).

The connection rate is the effective rate of dwords through the pathway between a SAS initiator phy and a SAS target phy, established through the connection request. Every phy shall support a 1,5 Gbps connection rate regardless of its ~~physical~~logical link rate.

No more than one connection is active on a ~~physical~~logical link at a time. If the connection is an SSP or SMP connection and there are no dwords to transmit associated with that connection, idle dwords are transmitted. If the connection is an STP connection and there are no dwords to transmit associated with that connection, SATA_SYNCs, SATA_CONTs, or vendor-specific scrambled data dwords (after a SATA_CONT) are transmitted. If there is no connection on a ~~physical~~logical link then idle dwords are transmitted.

The number of connections established by a SAS port shall not exceed the number of SAS phys within the SAS port (i.e., only one connection per SAS phy is allowed). There shall be a separate connection on each ~~physical~~logical link.

If multiple potential pathways exist between the SAS initiator port(s) and the SAS target port(s), multiple connections may be established by a SAS port between the following:

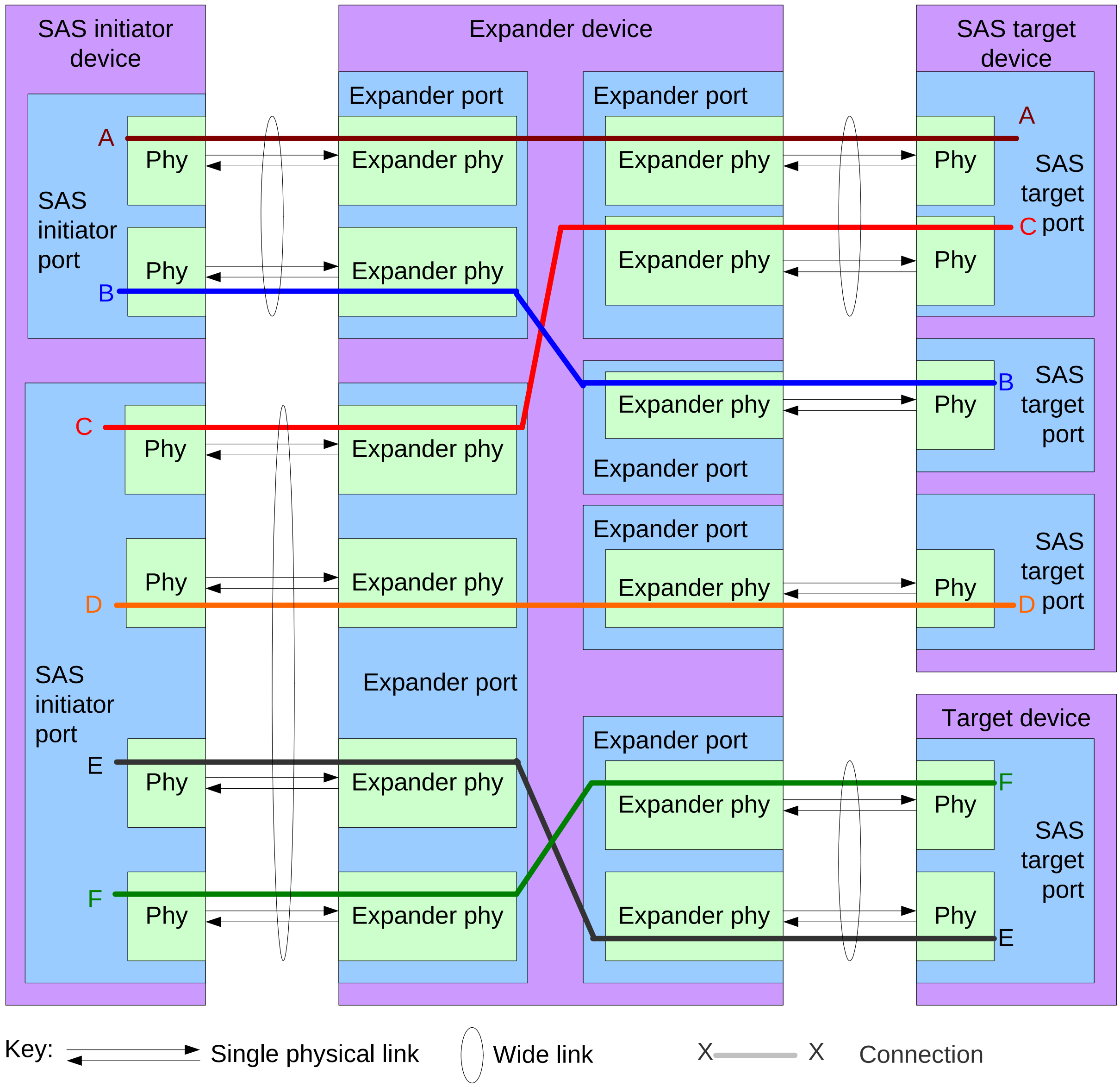
a) one SAS initiator port to multiple SAS target ports;
b) one SAS target port to multiple SAS initiator ports; or

c) one SAS initiator port to one SAS target port.

Once a connection is established, the pathway used for that connection shall not be changed (i.e., all the ~~physical~~logical links that make up the pathway remain dedicated to the connection until it is closed).

Figure 12 shows examples of connections between wide and narrow ports. All the connections shown may occur simultaneously. Additionally:

a) the connections labeled A and B are an example of one SAS initiator port with connections to multiple SAS target ports;
b) the connections labeled A and C are an example of one SAS target port with connections to multiple SAS initiator ports;
c) the connections labeled E and F are an example of multiple connections between one SAS initiator port and one SAS target port; and
d) the connections labeled C, D, E, and F are an example of one SAS initiator port with connections to multiple SAS target ports with one of those SAS target ports having multiple connections with that SAS initiator port.



Note: The expander device has a unique SAS address. Each SAS initiator port and SAS target port has a unique SAS address. Connections E and F represent a wide SAS initiator port with two simultaneous connections to a wide SAS target port.

Figure 12 — Multiple connections on wide ports [no update]

4.4.1 Reset overview

Figure 13 illustrates the reset terminology used in this standard:

a) link reset sequence;
b) phy reset sequence (see 6.7);
c) SATA OOB sequence (see 6.7.2.1);
d) SATA speed negotiation sequence (see 6.7.2.2);
e) SAS OOB sequence (see 6.7.4.1);
f) SAS speed negotiation sequence (see 6.7.4.2);
g) hard reset sequence (see 7.9);
h) identification sequence (see 7.9); and
i) multiplexing sequence (see 7.x).

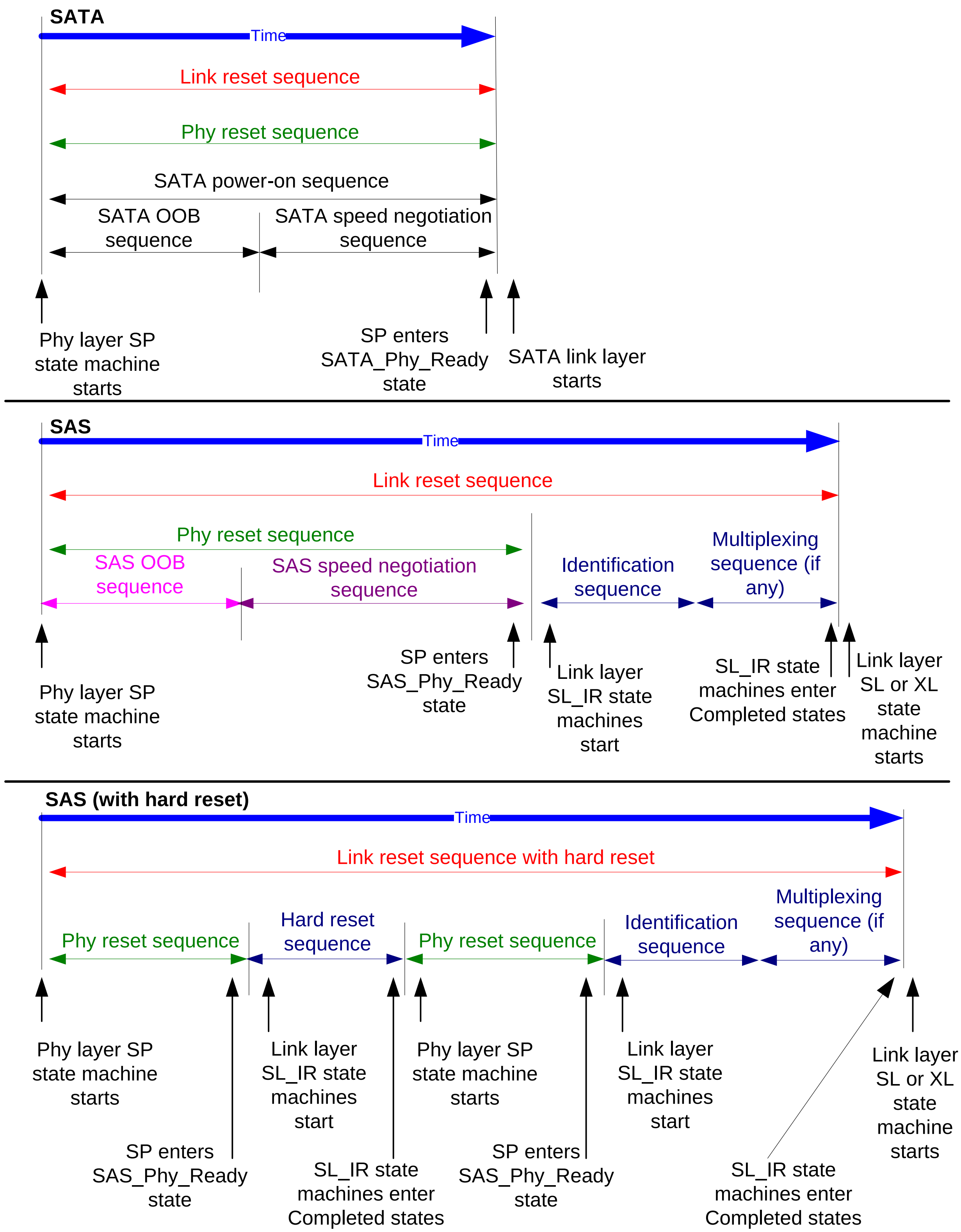


Figure 13 — Reset terminology [updated to add multiplexing sequence]

The phy reset sequences, including the OOB sequence and speed negotiation sequences, are implemented by the SP state machine and are described in 6.7 and 6.8. The hard reset sequence and identification sequence are implemented by the SL_IR state machine and are described in 7.9.

The link reset sequence has no effect on the transport layer and application layer. The HARD_RESET primitive sequence may be used during the identification sequence to initiate a hard reset. The link reset sequence serves as a hard reset for SATA devices.

4.6.7.2 Connection request routing

The ECM shall determine how to route a connection request from a source expander phy to a destination expander phy in a different expander port if the destination expander phy is enabled and operating at a valid physical link rate (e.g., the DISCOVER function reports a NEGOTIATED PHYSICAL LINK RATE field set to G1 (i.e., 8h) or G2 (i.e., 9h)) using the following precedence:

1) route to an expander phy with the direct routing attribute or table routing attribute when the destination SAS address matches the attached SAS address;
2) route to an expander phy with the table routing attribute when the destination SAS address matches an enabled SAS address in the expander route table;
3) route to an expander phy with the subtractive routing attribute; or
4) return an Arb Reject confirmation (see 4.6.6.3) to the source expander phy.

If the destination expander phy only matches an expander phy in the same expander port from which the connection request originated, then the ECM shall return an Arb Reject confirmation.

If the destination SAS address of a connection request matches a disabled SAS address in an expander route table, then the ECM shall ignore the match.

An expander phy that is multiplexing supports multiple connections at the same time, each with a connection rate limited by the logical link rate.

Editor's Note 2: more changes may be needed here to clarify that logical phys are the ones that communicate with the ECM.

4.7 Discover process

4.7.x Enabling multiplexing

Self-configuring expander devices shall configure multiplexing for their own phys. Any management application client may configure multiplexing for phys in non-self-configuring expander devices.

If the SAS domain contains all 6 Gbps target phys, then the management application clients should disable multiplexing on every phy.

If the SAS domain contains all 3 Gbps target phys, then the management application clients should:

a) multiplex 6 Gbps physical links into 3 Gbps logical links;
b) not multiplex 3 Gbps physical links.

If the SAS domain contains all 1,5 Gbps target phys, then the management application client should:

a) multiplex 6 Gbps physical links into four 1,5 Gbps logical links;
b) multiplex 3 Gbps physical links into two 1,5 Gbps logical links;

If the SAS domain contains a mixture of target phys with different physical link rates, then the management application client should multiplex physical links between initiator phys and target phys that request specific bandwidth in the REQUESTED BANDWIDTH field in the IDENTIFY address frame into logical links supporting connection rates supporting the lower of the requested bandwidth and the maximum connection rate of the pathway to the target phy.

The algorithm is as follows:

1) 6 Gbps allocations:
    1) Multiplex 6 Gbps physical links into 6 Gbps logical links (i.e., no multiplexing) and mark the logical links available;
    2) Allocate 6 Gbps logical links to target phys requesting 300 MBps or greater bandwidth;

        3) Allocate any unused bandwidth in the 6 Gbps logical links to target phys requesting that bandwidth;
2) 3 Gbps allocations:
    1) Multiplex any unallocated 6 Gbps physical links into 3 Gbps logical links; [favor 1:2 over 1:4]
    2) Allocate 3 Gbps logical links to target phys requesting 150 MBps or greater bandwidth;
    3) Allocate any unused bandwidth in the 3 Gbps logical links to target phys requesting that bandwidth;
    4) Multiplex 3 Gbps physical links into 3 Gbps logical links (i.e., no multiplexing);
    5) Allocate 3 Gbps logical links to target phys requesting 150 MBps or greater bandwidth;
    6) Allocate any unused bandwidth in the 3 Gbps logical links to target phys requesting that bandwidth;
3) 1,5 Gbps allocations: [at the final rate, there's no point in multiplexing 3 Gbps into 1.5 Gbps links and allocating to them before 1.5 physical links, since 1.5 Gbps cannot be multiplexed. So, allocate the 1.5 physical links first, then 3 Gbps multiplexed, then finally 6 Gbps multiplexed. If there is leftover bandwidth, it will sit at the highest rates rather than the lowest rates.]
    1) Multiplex 1,5 Gbps physical links into 1,5 Gbps logical links (i.e., no multiplexing);
    2) Allocate 1,5 Gbps logical links to target phys;
    3) Allocate any unused bandwidth in the 1,5 Gbps logical links to target phys requesting that bandwidth;
    4) Multiplex any unallocated 3 Gbps physical links into 1,5Gbps logical links; [favor 1:2 over 1:4]
    5) Allocate 1,5 Gbps logical links to target phys;
    6) Allocate any unused bandwidth in the 1,5 Gbps logical links to target phys requesting that bandwidth;
    7) Multiplex any unallocated 6 Gbps physical links into 1,5Gbps logical links; [finally resort to 1:4]
    8) Allocate 1,5 Gbps logical links to target phys;
    9) Allocate any unused bandwidth in the 1,5 Gbps logical links to target phys requesting that bandwidth;

NOTE 1 - This algorithm favors a small number of phys requesting high bandwidths over a large number phys requesting low bandwidths.

Editor's Note 3: This is the most problematic area for static multiplexing.

Editor's Note 4: The algorithm must converge when multiple expanders in a pathway are making their decisions independently and at different times. When any physical link multiplexing increases, the pathways available to the target phys have a smaller connection rate, which may cause the multiplexing on other phys to also be increased. When multiplexing decreases, the pathways support larger connection rates, and multiplexing may be turned off on other phys. It is critical that the expanders not keep changing their multiplexing level, causing changes in others that loop back and cause them to change their level again. It is also critical that they are ensured of reaching the desired result, and are not able to get stuck with the wrong result.

Editor's Note 5: The algorithm must be well enough defined so non-self configuring expanders are programmed the same way by different management application clients.

Editor's Note 6: A user should also be able to override the automatic algorithm and lock down their choices. That mechanism is outside the scope of the standard.

Figure 14 shows some examples.

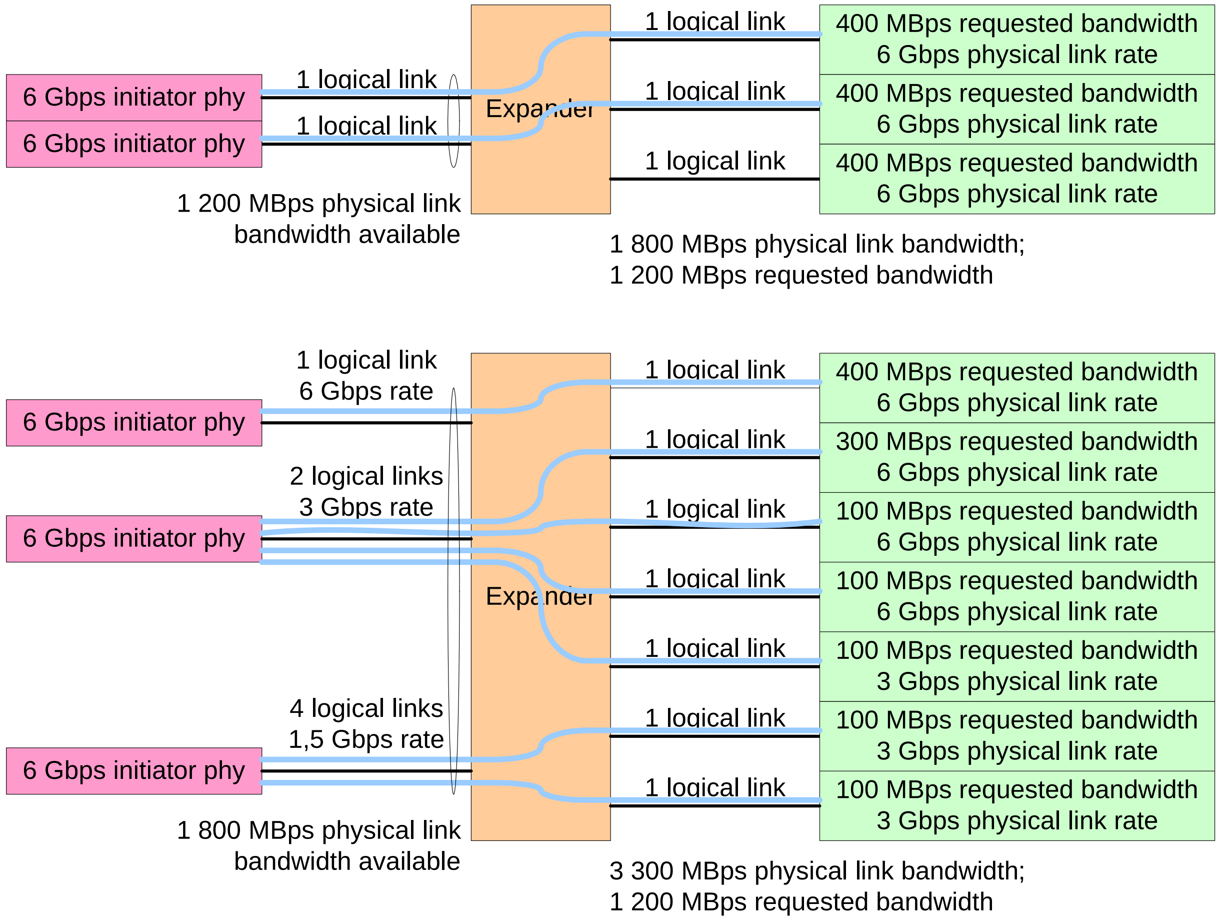


Figure 14 — Multiplexing bandwidth allocation

Changes to the phy layer

If multiplexing is enabled, the phy must give up immediately upon losing dword synchronization and not try to reacquire it, since it won't know which logical links are which.

6.8.4.9 SP15:SAS_PHY_Ready state

6.8.4.9.1 State description

This state waits for a COMINIT Detected message, a DWS Lost message, or a DWS Reset message.

While in this state dwords from the link layer are transmitted at the negotiated physical link rate at the rate established in the previous speed negotiation window.

Upon entry into this state, this state shall send a Phy Layer Ready (SAS) confirmation to the link layer to indicate that the physical link has been brought up successfully in SAS mode.

If the phy is not multiplexed into more than one logical phy, each ~~Each~~ time this state receives a DWS Lost message, this state may send a Start DWS message to the SP_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

6.8.4.9.2 Transition SP15:SAS_PHY_Ready to SP0:OOB_COMINIT

This transition shall occur after:

a) receiving a DWS Lost message, if this state does not send a Start DWS message;

b) receiving a DWS Lost message followed by a COMINIT Detected message, if this state does not send a Start DWS message; or
c) receiving a DWS Reset message.

This transition may but should not occur after receiving a COMINIT Detected message before receiving a DWS Lost message, or after receiving a COMINIT Detected message after sending a Start DWS message (i.e., the SP state machine should ignore COMINIT Detected messages unless the SP_DWS state machine has indicated loss of dword synchronization).

6.8.5.8 SP22:SATA_PHY_Ready state

6.8.5.8.1 State description

While in this state dwords from the link layer are transmitted at the negotiated physical link rate at the rate established in the previous speed negotiation window.

This state shall send a Phy Layer Ready (SATA) confirmation to the link layer to indicate that the physical link has been brought up successfully in SATA mode.

This state waits for a COMINIT Detected message, a DWS Lost message, or a DWS Reset message.

If the phy is not multiplexed into more than one logical phy, each ~~Each~~ time this state receives a DWS Lost message, this state may send a Start DWS message to the SP_DWS state machine to re-acquire dword synchronization without running a new link reset sequence.

6.8.5.8.2 Transition SP22:SATA_PHY_Ready to SP0:OOB_COMINIT

This transition shall occur after:

a) receiving a DWS Lost message, if this state does not send a Start DWS message;
a) receiving a DWS Lost message followed by a COMINIT Detected message, if this state does not send a Start DWS message; or
b) receiving a DWS Reset message.

This transition may but should not occur after receiving a COMINIT Detected message before receiving a DWS Lost message, or after receiving a COMINIT Detected message after sending a Start DWS message (i.e., the SP state machine should ignore COMINIT Detected messages unless the SP_DWS state machine has indicated loss of dword synchronization).

6.8.5.8.3 Transition SP22:SATA_PHY_Ready to SP23:SATA_PM_Partial

This transition shall occur after receiving an Enter Partial request.

6.8.5.8.4 Transition SP22:SATA_PHY_Ready to SP24:SATA_PM_Slumber

This transition shall occur after receiving an Enter Slumber request.

Changes to the link layer

Define the multiplexing sequence and the MUX primitives.

7.2.2 Primitive summary

Table 2 defines the primitives not specific to the type of connection.

Table 2 — Primitives not specific to type of connection (part 1 of 2)

Primitive	Use [a]	From [b]			To [b]			Primitive sequence type [c]
		I	E	T	I	E	T	
AIP (NORMAL)	NoConn		E		I	E	T	Single
AIP (RESERVED 0)	NoConn				I	E	T	Single
AIP (RESERVED 1)	NoConn				I	E	T	Single
AIP (RESERVED 2)	NoConn				I	E	T	Single
AIP (RESERVED WAITING ON PARTIAL)	NoConn				I	E	T	Single
AIP (WAITING ON CONNECTION)	NoConn		E		I	E	T	Single
AIP (WAITING ON DEVICE)	NoConn		E		I	E	T	Single
AIP (WAITING ON PARTIAL)	NoConn		E		I	E	T	Single
ALIGN (0)	All	I	E	T	I	E	T	Single
ALIGN (1)	All	I	E	T	I	E	T	Single
ALIGN (2)	All	I	E	T	I	E	T	Single
ALIGN (3)	All	I	E	T	I	E	T	Single
BREAK	All	I	E	T	I	E	T	Redundant
BROADCAST (CHANGE)	NoConn	I	E		I	E	T	Redundant
BROADCAST (SES)	NoConn			T	I	E	T	Redundant
BROADCAST (RESERVED 1)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED 2)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED 3)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED 4)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED CHANGE 0)	NoConn				I	E	T	Redundant
BROADCAST (RESERVED CHANGE 1)	NoConn				I	E	T	Redundant
CLOSE (CLEAR AFFILIATION)	STP	I					T	Triple
CLOSE (NORMAL)	Conn	I		T	I		T	Triple
CLOSE (RESERVED 0)	Conn				I		T	Triple
CLOSE (RESERVED 1)	Conn				I		T	Triple
EOAF	NoConn	I	E	T	I	E	T	Single
ERROR	All		E		I	E	T	Single
HARD_RESET	NoConn	I	E		I	E	T	Redundant
MUX (0)	NoConn	I	E	T	I	E	T	Single
MUX (1)	NoConn	I	E	T	I	E	T	Single
MUX (2)	NoConn	I	E	T	I	E	T	Single
MUX (3)	NoConn	I	E	T	I	E	T	Single
NOTIFY (ENABLE SPINUP)	All	I	E				T	Single
NOTIFY (POWER FAILURE EXPECTED)	All	I	E				T	Single
NOTIFY (RESERVED 1)	All				I	E	T	Single
NOTIFY (RESERVED 2)	All				I	E	T	Single
OPEN_ACCEPT	NoConn	I		T	I		T	Single

Table 2 — Primitives not specific to type of connection (part 2 of 2)

Primitive	Use [a]	From [b]			To [b]			Primitive sequence type [c]
		I	E	T	I	E	T	
OPEN_REJECT (BAD DESTINATION)	NoConn		E		I		T	Single
OPEN_REJECT (CONNECTION RATE NOT SUPPORTED)	NoConn	I	E	T	I		T	Single
OPEN_REJECT (NO DESTINATION)	NoConn		E		I		T	Single
OPEN_REJECT (PATHWAY BLOCKED)	NoConn		E		I		T	Single
OPEN_REJECT (PROTOCOL NOT SUPPORTED)	NoConn	I		T	I		T	Single
OPEN_REJECT (RESERVED ABANDON 0)	NoConn				I		T	Single
OPEN_REJECT (RESERVED ABANDON 1)	NoConn				I		T	Single
OPEN_REJECT (RESERVED ABANDON 2)	NoConn				I		T	Single
OPEN_REJECT (RESERVED ABANDON 3)	NoConn				I		T	Single
OPEN_REJECT (RESERVED CONTINUE 0)	NoConn				I		T	Single
OPEN_REJECT (RESERVED CONTINUE 1)	NoConn				I		T	Single
OPEN_REJECT (RESERVED INITIALIZE 0)	NoConn				I		T	Single
OPEN_REJECT (RESERVED INITIALIZE 1)	NoConn				I		T	Single
OPEN_REJECT (RESERVED STOP 0)	NoConn				I		T	Single
OPEN_REJECT (RESERVED STOP 1)	NoConn				I		T	Single
OPEN_REJECT (RETRY)	NoConn	I		T	I		T	Single
OPEN_REJECT (STP RESOURCES BUSY)	NoConn		E	T	I			Single
OPEN_REJECT (WRONG DESTINATION)	NoConn	I		T	I		T	Single
SOAF	NoConn	I	E	T	I	E	T	Single

[a] The Use column indicates when the primitive is used:
a) NoConn: SAS physical links, outside connections;
b) Conn: SAS physical links, inside connections;
c) All: SAS physical links, both outside connections or inside any type of connection; or
d) STP: SAS physical links, inside STP connections.

[b] The From and To columns indicate the type of ports that originate each primitive or are the intended destinations of each primitive:
a) I for SAS initiator ports;
b) E for expander ports; and
c) T for SAS target ports.

Expander ports are not considered originators of primitives that are passing through from expander port to expander port.

[c] The Primitive sequence type columns indicate whether the primitive is sent as a single primitive sequence, a repeated primitive sequence, a continued primitive sequence, a triple primitive sequence, or a redundant primitive sequence (see 7.2.4).

7.2.3 Primitive encodings

Table 3 defines the primitive encoding for primitives not specific to type of connection.

Table 3 — Primitive encoding for primitives not specific to type of connection (part 1 of 2)

Primitive	**Character**			
	1st	**2nd**	**3rd**	**4th (last)**
AIP (NORMAL)	K28.5	D27.4	D27.4	D27.4
AIP (RESERVED 0)	K28.5	D27.4	D31.4	D16.7
AIP (RESERVED 1)	K28.5	D27.4	D16.7	D30.0
AIP (RESERVED 2)	K28.5	D27.4	D29.7	D01.4
AIP (RESERVED WAITING ON PARTIAL)	K28.5	D27.4	D01.4	D07.3
AIP (WAITING ON CONNECTION)	K28.5	D27.4	D07.3	D24.0
AIP (WAITING ON DEVICE)	K28.5	D27.4	D30.0	D29.7
AIP (WAITING ON PARTIAL)	K28.5	D27.4	D24.0	D04.7
ALIGN (0)	K28.5	D10.2	D10.2	D27.3
ALIGN (1)	K28.5	D07.0	D07.0	D07.0
ALIGN (2)	K28.5	D01.3	D01.3	D01.3
ALIGN (3)	K28.5	D27.3	D27.3	D27.3
BREAK	K28.5	D02.0	D24.0	D07.3
BROADCAST (CHANGE)	K28.5	D04.7	D02.0	D01.4
BROADCAST (SES)	K28.5	D04.7	D07.3	D29.7
BROADCAST (RESERVED 1)	K28.5	D04.7	D01.4	D24.0
BROADCAST (RESERVED 2)	K28.5	D04.7	D04.7	D04.7
BROADCAST (RESERVED 3)	K28.5	D04.7	D16.7	D02.0
BROADCAST (RESERVED 4)	K28.5	D04.7	D29.7	D30.0
BROADCAST (RESERVED CHANGE 0)	K28.5	D04.7	D24.0	D31.4
BROADCAST (RESERVED CHANGE 1)	K28.5	D04.7	D27.4	D07.3
CLOSE (CLEAR AFFILIATION)	K28.5	D02.0	D07.3	D04.7
CLOSE (NORMAL)	K28.5	D02.0	D30.0	D27.4
CLOSE (RESERVED 0)	K28.5	D02.0	D31.4	D30.0
CLOSE (RESERVED 1)	K28.5	D02.0	D04.7	D01.4
EOAF	K28.5	D24.0	D07.3	D31.4
ERROR	K28.5	D02.0	D01.4	D29.7
HARD_RESET	K28.5	D02.0	D02.0	D02.0
MUX (0)	K28.5	D02.0	D16.7	D31.4
MUX (1)	K28.5	D07.3	D04.7	D30.0
MUX (2)	K28.5	D16.7	D24.0	D27.4

Table 3 — Primitive encoding for primitives not specific to type of connection (part 2 of 2)

Primitive	Character			
	1st	2nd	3rd	4th (last)
MUX (3)	K28.5	D24.0	D01.4	D16.7
NOTIFY (ENABLE SPINUP)	K28.5	D31.3	D31.3	D31.3
NOTIFY (POWER FAILURE EXPECTED)	K28.5	D31.3	D07.0	D01.3
NOTIFY (RESERVED 1)	K28.5	D31.3	D01.3	D07.0
NOTIFY (RESERVED 2)	K28.5	D31.3	D10.2	D10.2
OPEN_ACCEPT	K28.5	D16.7	D16.7	D16.7
OPEN_REJECT (BAD DESTINATION)	K28.5	D31.4	D31.4	D31.4
OPEN_REJECT (CONNECTION RATE NOT SUPPORTED)	K28.5	D31.4	D04.7	D29.7
OPEN_REJECT (NO DESTINATION)	K28.5	D29.7	D29.7	D29.7
OPEN_REJECT (PATHWAY BLOCKED)	K28.5	D29.7	D16.7	D04.7
OPEN_REJECT (PROTOCOL NOT SUPPORTED)	K28.5	D31.4	D29.7	D07.3
OPEN_REJECT (RESERVED ABANDON 0)	K28.5	D31.4	D02.0	D27.4
OPEN_REJECT (RESERVED ABANDON 1)	K28.5	D31.4	D30.0	D16.7
OPEN_REJECT (RESERVED ABANDON 2)	K28.5	D31.4	D07.3	D02.0
OPEN_REJECT (RESERVED ABANDON 3)	K28.5	D31.4	D01.4	D30.0
OPEN_REJECT (RESERVED CONTINUE 0)	K28.5	D29.7	D02.0	D30.0
OPEN_REJECT (RESERVED CONTINUE 1)	K28.5	D29.7	D24.0	D01.4
OPEN_REJECT (RESERVED INITIALIZE 0)	K28.5	D29.7	D30.0	D31.4
OPEN_REJECT (RESERVED INITIALIZE 1)	K28.5	D29.7	D07.3	D16.7
OPEN_REJECT (RESERVED STOP 0)	K28.5	D29.7	D31.4	D07.3
OPEN_REJECT (RESERVED STOP 1)	K28.5	D29.7	D04.7	D27.4
OPEN_REJECT (RETRY)	K28.5	D29.7	D27.4	D24.0
OPEN_REJECT (STP RESOURCES BUSY)	K28.5	D31.4	D27.4	D01.4
OPEN_REJECT (WRONG DESTINATION)	K28.5	D31.4	D16.7	D24.0
SOAF	K28.5	D24.0	D30.0	D01.4

7.2.5.n MUX (Multiplex)

MUX is sent by a phy to negotiate multiplexing.

The versions of MUX are defined in table 4.

Table 4 — MUX primitives

Primitive	Description
MUX (0)	Establishes the position of dwords in logical link 0.
MUX (1)	Establishes the position of dwords in logical link 1.
MUX (2)	Establishes the position of dwords in logical link 0 or 2.
MUX (3)	Establishes the position of dwords in logical link 1 or 3.

Phys shall rotate through MUX (0), MUX (1), MUX (2), and MUX (3) for all MUXs transmitted after the identification sequence.

See 7.12 for details on connections.

7.8.2 IDENTIFY address frame

Table 5 defines the IDENTIFY address frame format used for the identification sequence. The IDENTIFY address frame is sent after the phy reset sequence completes if the physical link is a SAS physical link.

Table 5 — IDENTIFY address frame format

Byte\Bit	7	6	5	4	3	2	1	0
0	Restricted (for OPEN address frame)	DEVICE TYPE			ADDRESS FRAME TYPE (0h)			
1	Reserved		REQUESTED LOGICAL LINKS		Restricted (for OPEN address frame)			
2	Reserved				SSP INITIATOR PORT	STP INITIATOR PORT	SMP INITIATOR PORT	Restricted (for OPEN address frame)
3	Reserved				SSP TARGET PORT	STP TARGET PORT	SMP TARGET PORT	Restricted (for OPEN address frame)
4	Restricted (for OPEN address frame)							
11								
12	SAS ADDRESS							
19								
20	PHY IDENTIFIER							
20	REQUESTED BANDWIDTH							
21	Reserved							
27								
28	(MSB)	CRC						
31								(LSB)

The DEVICE TYPE field specifies the type of device containing the phy, and is defined in table 6.

Table 6 — DEVICE TYPE field

Code	Description
001b	End device
010b	Edge expander device
011b	Fanout expander device
All others	Reserved

The ADDRESS FRAME TYPE field shall be set to 0h.

The REQUESTED LOGICAL LINKS field indicates the maximum number of logical links the phy supports via multiplexing and is defined in table 22.

Table 7 — REQUESTED LOGICAL LINKS field

Code	Physical link rate		
	6 Gbps	**3 Gbps**	**1,5 Gbps**
00b	One logical link (i.e., no multiplexing)		
01b	Two logical links		One logical link
10b	Four logical links	Two logical links	
11b	Reserved		

If the phy is controlled by an SMP target port, the REQUESTED LOGICAL LINKS field is based on the the DESIRED LOGICAL LINK RATE field in the SMP PHY CONTROL and DISCOVER functions as described in table 22.

Table 8 — REQUESTED LOGICAL LINKS field

DESIRED LOGICAL LINK RATE field in DISCOVER	Physical link rate	Resulting REQUESTED LOGICAL LINKS field
6 Gbps	1,5 Gbps	00b, 01b, or 10b (i.e., one 1,5 Gbps logical link)
	3 Gbps	00b or 01b (i.e., one 3 Gbps logical link)
	6 Gbps	00b (i.e., one 6 Gbps logical link)
3 Gbps	1,5 Gbps	00b, 01b, or 10b (i.e., one 1,5 Gbps logical link)
	3 Gbps	00b (i.e., two 1,5 Gbps logical links)
	6 Gbps	01b or 10b (i.e., two 3 Gbps logical links)
1,5 Gbps	1,5 Gbps	00b, 01b, or 10b (i.e., one 1,5 Gbps logical link)
	3 Gbps	01b or 10b (i.e., two 1,5 Gbps logical links)
	6 Gbps	10b (i.e., four 1,5 Gbps logical links)

The REQUESTED BANDWIDTH field indicates the amount of sustained bandwidth, in units of 10 MBps, a target phy is capable of transmitting or receiving, whichever is higher. A REQUESTED BANDWIDTH field set to 00h indicates the amount of bandwidth is unknown. This field shall be set to 00h by phys that are only used by initiator ports or by SMP ports. This field should only be set to a non-zero value if the target phy needs connections to use a minimum connection rate to avoid performance problems (e.g., to keep tape drives streaming and to avoid rotational latency in disk drives during sequential operations).

Examples:

a) a disk drive with an outer track media rate of 60 MBps should set this field to 6;
b) a tape drive with an expected compressed transfer rate of 320 MBps should set this field to 32; and
c) a SAS-attached RAID controller with a wide port providing access to a logical unit built from eight striped 60 MBps disk drives should set this field to 48.

This field should not reflect higher bandwidth capabilities that are limited in duration (e.g., accesses to cache memory). This field is used by management application clients to configure multiplexing (see 7.xx). If the phy is part of a wide SSP port, or the phy is expected to become part of a wide SSP port but the link reset sequences have not completed on the other phys in the wide port yet, and the SSP port supports multiple commands outstanding, this field should be set to the bandwidth that the phy is capable of delivering for a single command.

An SSP INITIATOR PORT bit set to one ~~specifies~~indicates that an SSP initiator port is present. An SSP INITIATOR PORT bit set to zero ~~specifies~~indicates that an SSP initiator port is not present. Expander devices shall set the SSP INITIATOR PORT bit to zero.

An STP INITIATOR PORT bit set to one ~~specifies~~indicates that an STP initiator port is present. An STP INITIATOR PORT bit set to zero ~~specifies~~indicates that an STP initiator port is not present. Expander devices shall set the STP INITIATOR PORT bit to zero.

An SMP INITIATOR PORT bit set to one ~~specifies~~indicates that an SMP initiator port is present. An SMP INITIATOR PORT bit set to zero ~~specifies~~indicates that an SMP initiator port is not present. Expander devices may set the SMP INITIATOR PORT bit to one.

An SSP TARGET PORT bit set to one ~~specifies~~indicates that an SSP target port is present. An SSP TARGET PORT bit set to zero ~~specifies~~indicates that an SSP target port is not present. Expander devices shall set the SSP TARGET PORT bit to zero.

An STP TARGET PORT bit set to one ~~specifies~~indicates that an STP target port is present. An STP TARGET PORT bit set to zero ~~specifies~~indicates that an STP target port is not present. Expander devices shall set the STP TARGET PORT bit to zero.

An SMP TARGET PORT bit set to one ~~specifies~~indicates that an SMP target port is present. An SMP TARGET PORT bit set to zero ~~specifies~~indicates that an SMP target port is not present. Expander devices shall set the SMP TARGET PORT bit to one.

For SAS ports, the SAS ADDRESS field ~~specifies~~indicates the port identifier (see 4.2.6) of the SAS port transmitting the IDENTIFY address frame. For expander ports, the SAS ADDRESS field ~~specifies~~indicates the device name (see 4.2.4) of the expander device transmitting the IDENTIFY address frame.

The PHY IDENTIFIER field ~~specifies~~indicates the phy identifier of the phy transmitting the IDENTIFY address frame.

See 4.1.3 for additional requirements concerning the DEVICE TYPE field, SSP INITIATOR PORT bit, STP INITIATOR PORT bit, SMP INITIATOR PORT bit, SSP TARGET PORT bit, STP TARGET PORT bit, SMP TARGET PORT bit, and SAS ADDRESS field.

The CRC field is defined in 7.8.1.

7.9 Identification and hard reset sequence

7.9.1 Identification and hard reset sequence overview

After the phy reset sequence has been completed indicating the physical link is using SAS rather than SATA, each phy transmits either:

a) an IDENTIFY address frame (see); or
b) a HARD_RESET primitive sequence.

Each phy receives an IDENTIFY address frame or a HARD_RESET primitive sequence from the phy to which it is attached. The combination of a phy reset sequence, an optional hard reset sequence, and an identification sequence is called a link reset sequence (see 4.4.1).

If a phy receives a valid IDENTIFY address frame within 1 ms of phy reset sequence completion, the SAS address in the outgoing IDENTIFY address frame and the SAS address in the incoming IDENTIFY address frame determine the port to which a phy belongs (see 4.1.3). The phy ignores subsequent IDENTIFY address frames and HARD_RESET primitives until another phy reset sequence occurs.

If a phy receives a HARD_RESET primitive sequence within 1 ms of phy reset sequence completion, it shall be considered a reset event and cause a hard reset (see 4.4.2) of the port containing that phy.

If a phy does not receive a HARD_RESET primitive sequence or a valid IDENTIFY address frame within 1 ms of phy reset sequence completion, it shall restart the phy reset sequence.

7.9.2 SAS initiator device rules

After a link reset sequence, or after receiving a BROADCAST (CHANGE), a management application client behind an SMP initiator port should perform a discover process (see 4.7).

When a discover process is performed after a link reset sequence, the management application client discovers all the devices in the SAS domain. When a discover process is performed after a BROADCAST (CHANGE), the management application client determines which devices have been added to or removed from the SAS domain.

The discover information may be used to select connection rates for connection requests (see 7.8.3).

7.9.3 Fanout expander device rules

After completing the identification sequence on a phy and completing internal initialization, the ECM within a fanout expander device shall be capable of routing connection requests through that phy. The expander device may return OPEN_REJECT (NO DESTINATION) until it is ready to process connection requests.

After a link reset sequence, or after receiving a BROADCAST (CHANGE), the management application client behind an SMP initiator port in a fanout expander device that does not have a configurable expander route table shall follow the SAS initiator device rules (see 7.9.2) to perform a discover process.

The ECM of a fanout expander device that has a configurable expander route table is dependent on the completion of the discover process (see 4.7) for routing connection requests using the table routing method.

7.9.4 Edge expander device rules

After completing the identification sequence on a phy and completing internal initialization, the ECM within an edge expander device shall be capable of routing connection requests through that phy. The expander device may return OPEN_REJECT (NO DESTINATION) until it is ready to process connection requests.

The ECM of an edge expander device that has a configurable expander route table is dependent on the completion of the discover process (see 4.7) for routing connection requests using the table routing method.

7.9.5 SL_IR (link layer identification and hard reset) state machines

7.9.5.1 SL_IR state machines overview

The SL_IR (link layer identification and hard reset) state machines control the flow of dwords on the physical link that are associated with the identification and hard reset sequences. The state machines are as follows:

a) SL_IR_TIR (transmit IDENTIFY or HARD_RESET) state machine (see 7.9.5.3);
b) SL_IR_RIF (receive IDENTIFY address frame) state machine (see 7.9.5.4); and
c) SL_IR_IRC (identification and hard reset control) state machine (see 7.9.5.5).

The SL_IR state machines send the following messages to the SL state machines (see 7.14) in SAS devices or the XL (see 7.15) state machine in expander devices:

a) Enable Disable SAS Link (Enable); and
b) Enable Disable SAS Link (Disable).

The SL_IR_IRC state machine shall maintain the timers listed in table 9.

Table 9 — SL_IR_IRC timers

Timer	Initial value
Receive Identify Timeout timer	1 ms

Figure 15 shows the SL_IR state machines.

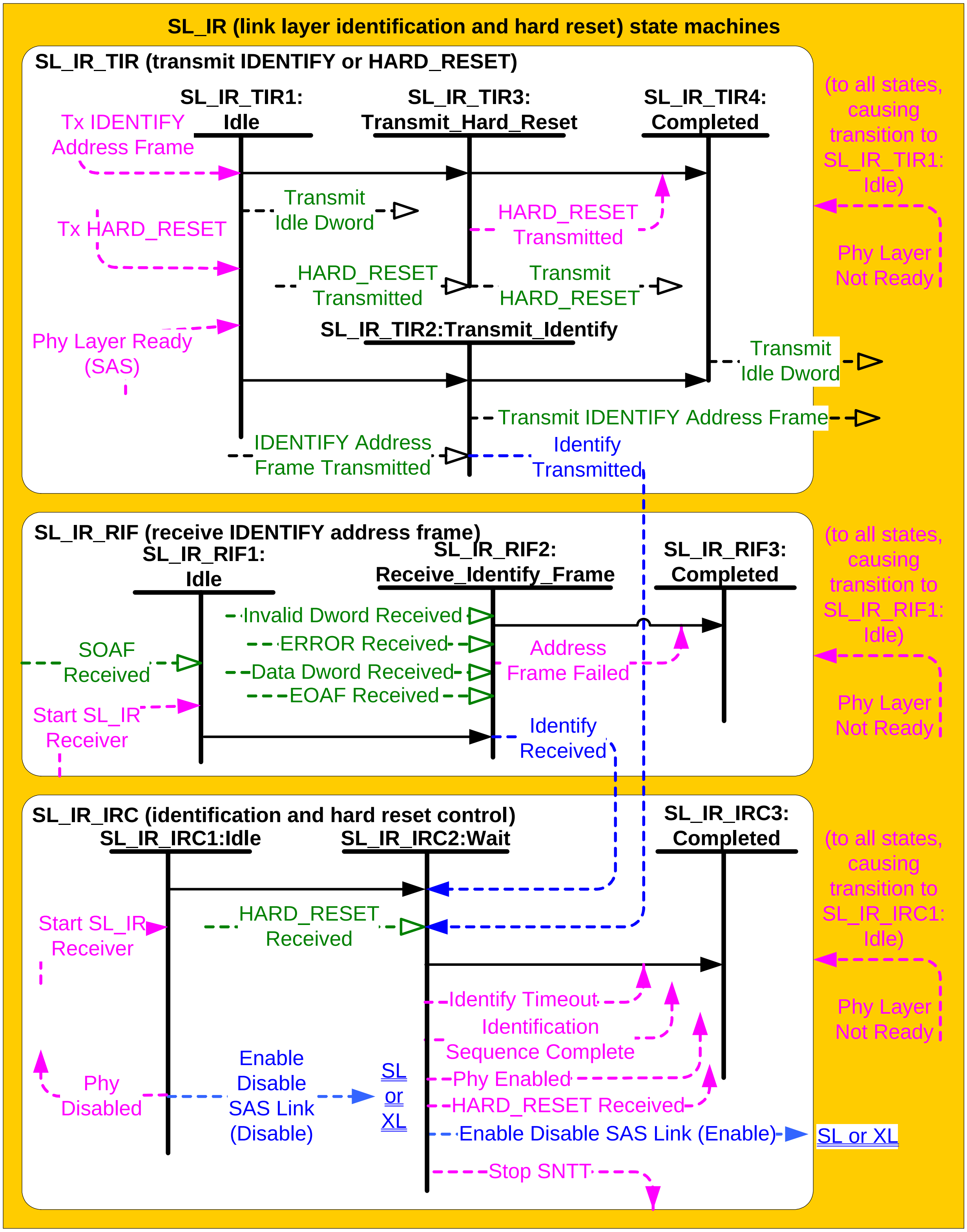


Figure 15 — SL_IR (link layer identification and hard reset) state machines

7.9.5.2 SL_IR transmitter and receiver

The SL_IR transmitter receives the following messages from the SL_IR state machines indicating primitive sequences, frames, and dwords to transmit:

a) Transmit IDENTIFY Address Frame;
b) Transmit HARD_RESET; and
c) Transmit Idle Dword.

The SL_IR transmitter sends the following messages to the SL_IR state machines:

a) HARD_RESET Transmitted; and
b) IDENTIFY Address Frame Transmitted.

The SL_IR receiver sends the following messages to the SL_IR state machines indicating primitive sequences and dwords received from the SP_DWS receiver (see 6.9.2):

a) SOAF Received;
b) Data Dword Received;
c) EOAF Received;
d) ERROR Received;
e) Invalid Dword Received; and
f) HARD_RESET Received.

The SL_IR receiver shall ignore all other dwords.

7.9.5.3 SL_IR_TIR (transmit IDENTIFY or HARD_RESET) state machine

7.9.5.3.1 SL_IR_TIR state machine overview

The SL_IR_TIR state machine's function is to transmit a single IDENTIFY address frame or HARD_RESET primitive after the phy layer enables the link layer. This state machine consists of the following states:

a) SL_IR_TIR1:Idle (see 7.9.5.3.2)(initial state);
b) SL_IR_TIR2:Transmit_Identify (see 7.9.5.3.3);
c) SL_IR_TIR3:Transmit_Hard_Reset (see 7.9.5.3.4); and
d) SL_IR_TIR4:Completed (see 7.9.5.3.5).

This state machine shall start in the SL_IR_TIR1:Idle state. This state machine shall transition to the SL_IR_TIR1:Idle state from any other state after receiving a Phy Layer Not Ready confirmation.

7.9.5.3.2 SL_IR_TIR1:Idle state

7.9.5.3.2.1 State description

This state shall request idle dwords be transmitted by repeatedly sending Transmit Idle Dword messages to the SL_IR transmitter.

7.9.5.3.2.2 Transition SL_IR_TIR1:Idle to SL_IR_TIR2:Transmit_Identify

This transition shall occur after both:

a) a Phy Layer Ready (SAS) confirmation is received; and
b) a Tx IDENTIFY Address Frame request is received.

7.9.5.3.2.3 Transition SL_IR_TIR1:Idle to SL_IR_TIR3:Transmit_Hard_Reset

This transition shall occur after both:

a) a Phy Layer Ready (SAS) confirmation is received; and
b) a Tx HARD_RESET request is received.

7.9.5.3.3 SL_IR_TIR2:Transmit_Identify state

7.9.5.3.3.1 State description

Upon entry into this state, this state shall send a Transmit IDENTIFY Address Frame message to the SL_IR transmitter.

After this state receives an IDENTIFY Address Frame Transmitted message, this state shall send an Identify Transmitted message to the SL_IR_IRC state machine.

7.9.5.3.3.2 Transition SL_IR_TIR2:Transmit_Identify to SL_IR_TIR4:Completed

This transition shall occur after sending an Identify Transmitted message to the SL_IR_IRC state machine.

7.9.5.3.4 SL_IR_TIR3:Transmit_Hard_Reset state

7.9.5.3.4.1 State description

Upon entry into this state, this state shall send a Transmit HARD_RESET message to the SL_IR transmitter.

After this state receives a HARD_RESET Transmitted message, this state shall send a HARD_RESET Transmitted confirmation to the management application layer.

7.9.5.3.4.2 Transition SL_IR_TIR3:Transmit_Hard_Reset to SL_IR_TIR4:Completed

This transition shall occur after sending a HARD_RESET Transmitted confirmation to the management application layer.

Editor’s Note 7: Consider adding state(s) to handle the multiplexing sequence

7.9.5.3.5 SL_IR_TIR4:Completed state

This state shall request idle dwords be transmitted by repeatedly sending Transmit Idle Dword messages to the SL_IR transmitter.

7.9.5.4 SL_IR_RIF (receive IDENTIFY address frame) state machine

7.9.5.4.1 SL_IR_RIF state machine overview

The SL_IR_RIF state machine receives an IDENTIFY address frame and checks the IDENTIFY address frame to determine if the frame should be accepted or discarded by the link layer.

This state machine consists of the following states:

a) SL_IR_RIF1:Idle (see 7.9.5.4.2)(initial state);
b) SL_IR_RIF2:Receive_Identify_Frame (see 7.9.5.4.3); and
c) SL_IR_RIF3:Completed (see 7.9.5.4.4).

This state machine shall start in the SL_IR_RIF1:Idle state. This state machine shall transition to the SL_IR_RIF1:Idle state from any other state after receiving a Phy Layer Not Ready confirmation.

7.9.5.4.2 SL_IR_RIF1:Idle state

7.9.5.4.2.1 State description

This state waits for an SOAF to be received from the physical link, indicating an address frame is arriving.

7.9.5.4.2.2 Transition SL_IR_RIF1:Idle to SL_IR_RIF2:Receive_Identify_Frame

This transition shall occur after both:

a) a Start SL_IR Receiver confirmation is received; and

b) an SOAF Received message is received.

7.9.5.4.3 SL_IR_RIF2:Receive_Identify_Frame state

7.9.5.4.3.1 State description

This state receives the dwords of an address frame and the EOAF.

If this state receives an SOAF Received message, then this state shall discard the address frame (i.e., the subsequent Data Dword Received and EOAF Received messages) and send an Address Frame Failed confirmation to the management application layer to indicate that an invalid IDENTIFY address frame was received.

If this state receives more than eight Data Dword Received messages after an SOAF Received message and before an EOAF Received message, then this state shall discard the address frame and send an Address Frame Failed confirmation to the management application layer to indicate that an invalid IDENTIFY address frame was received.

If this state receives an Invalid Dword Received message or an ERROR Received message after an SOAF Received message and before an EOAF Received message, then this state shall:

a) ignore the invalid dword or ERROR; or
b) discard the address frame and send an Address Frame Failed confirmation to the management application layer to indicate that an invalid IDENTIFY address frame was received.

After receiving an EOAF Received message, this state shall check if it the received frame is a valid IDENTIFY address frame.

This state shall accept an IDENTIFY address frame and send an Identify Received message to the SL_IR_IRC state machine if:

a) the ADDRESS FRAME TYPE field is set to Identify;
b) the number of bytes between the SOAF and EOAF is 32; and
c) the CRC field contains a valid CRC.

Otherwise, this state shall discard the IDENTIFY address frame and send an Address Frame Failed confirmation to the management application layer to indicate that an invalid IDENTIFY address frame was received.

7.9.5.4.3.2 Transition SL_IR_RIF2:Receive_Identify_Frame to SL_IR_RIF3:Completed

This transition shall occur after sending an Identify Received message or Address Frame Failed confirmation.

Editor's Note 8: Consider adding state(s) to handle the multiplexing sequence

---

7.9.5.4.4 SL_IR_RIF3:Completed state

This state waits for a Phy Layer Not Ready confirmation.

7.9.5.5 SL_IR_IRC (identification and hard reset control) state machine

7.9.5.5.1 SL_IR_IRC state machine overview

The SL_IR_IRC state machine ensures that IDENTIFY address frames have been both received and transmitted before enabling the rest of the link layer, and notifies the link layer if a HARD_RESET primitive sequence is received before an IDENTIFY address frame has been received.

This state machine consists of the following states:

a) SL_IR_IRC1:Idle (see 7.9.5.5.2)(initial state);
b) SL_IR_IRC2:Wait (see 7.9.5.5.3); and
c) SL_IR_IRC3:Completed (see 7.9.5.5.4).

This state machine shall start in the SL_IR_IRC1:Idle state. This state machine shall transition to the SL_IR_IRC1:Idle state from any other state after receiving a Phy Layer Not Ready confirmation.

7.9.5.5.2 SL_IR_IRC1:Idle state

7.9.5.5.2.1 State description

This state waits for the link layer to be enabled. Upon entry into this state, this state shall:

a) send an Enable Disable SAS Link (Disable) message to SL state machines (see 7.14) or XL state machine (see 7.15) halting any link layer activity; and
b) send a Phy Disabled confirmation to the port layer and the management application layer indicating that the phy is not ready for use.

7.9.5.5.2.2 Transition SL_IR_IRC1:Idle to SL_IR_IRC2:Wait

This transition shall occur after a Start SL_IR Receiver confirmation is received.

7.9.5.5.3 SL_IR_IRC2:Wait state

7.9.5.5.3.1 State description

This state ensures that an IDENTIFY address frame has been received by the SL_IR_RIF state machine and that a IDENTIFY address frame has been transmitted by the SL_IR_TIR state machine before enabling the rest of the link layer. The IDENTIFY address frames may be transmitted and received on the physical link in any order.

After this state receives an Identify Received message, it shall send a Stop SNTT request to the phy layer.

After this state receives an Identify Transmitted message, it shall initialize and start the Receive Identify Timeout timer. If an Identify Received message is received before the Receive Identify Timeout timer expires, this state shall:

a) send an Identification Sequence Complete confirmation to the management application layer, with arguments carrying the contents of the incoming IDENTIFY address frame;
b) send an Enable Disable SAS Link (Enable) message to the SL state machines (see 7.14) in a SAS phy or the XL state machine (see 7.15) in an expander phy indicating that the rest of the link layer may start operation; and
c) send a Phy Enabled confirmation to the port layer and the management application layer indicating that the phy is ready for use.

If the Receive Identify Timeout timer expires before an Identify Received message is received, this state shall send an Identify Timeout confirmation to the management application layer to indicate that an identify timeout occurred.

If this state receives a HARD_RESET Received message before an Identify Received message is received, this state shall send a HARD_RESET Received confirmation to the port layer and a Stop SNTT request to the phy layer.

If this state receives a HARD_RESET Received message after an Identify Received message is received, the HARD_RESET Received message shall be ignored.

7.9.5.5.3.2 Transition SL_IR_IRC2:Wait to SL_IR_IRC3:Completed

This transition shall occur after sending a HARD_RESET Received confirmation, Identify Timeout confirmation, or an Identification Sequence Complete and an Phy Enabled confirmation.

Editor’s Note 9: Consider adding state(s) to handle the multiplexing sequence

---

7.9.5.5.4 SL_IR_IRC3:Completed state

This state waits for a Phy Layer Not Ready confirmation.

7.xx Multiplexing

If a phy both transmits and receives an IDENTIFY address frame indicating that multiplexing is supported, it shall enable multiplexing to the highest common number of logical links after the identification sequence completes. This is called the multiplexing sequence.

The phy shall ignore all incoming dwords except MUX primitives. Incoming MUXes are not accompanied by ALIGNs and/or NOTIFYs, so the phy shall process them in logic running off the received clock, without using an elasticity buffer.

The phy shall transmit MUX repeatedly, rotating through MUX (0), MUX (1), MUX (2), and MUX (3) in order. The phy shall not transmit ALIGNs and/or NOTIFYs during the multiplexing sequence.

After the phy receives at least 3 MUX primitives confirming the position of dwords in each logical link, it shall continue transmitting at least 24 MUX primitives. The phy shall then stop transmitting MUX and the logical phys shall start transmitting dwords for the logical links in the corresponding positions as shown in figure 16.

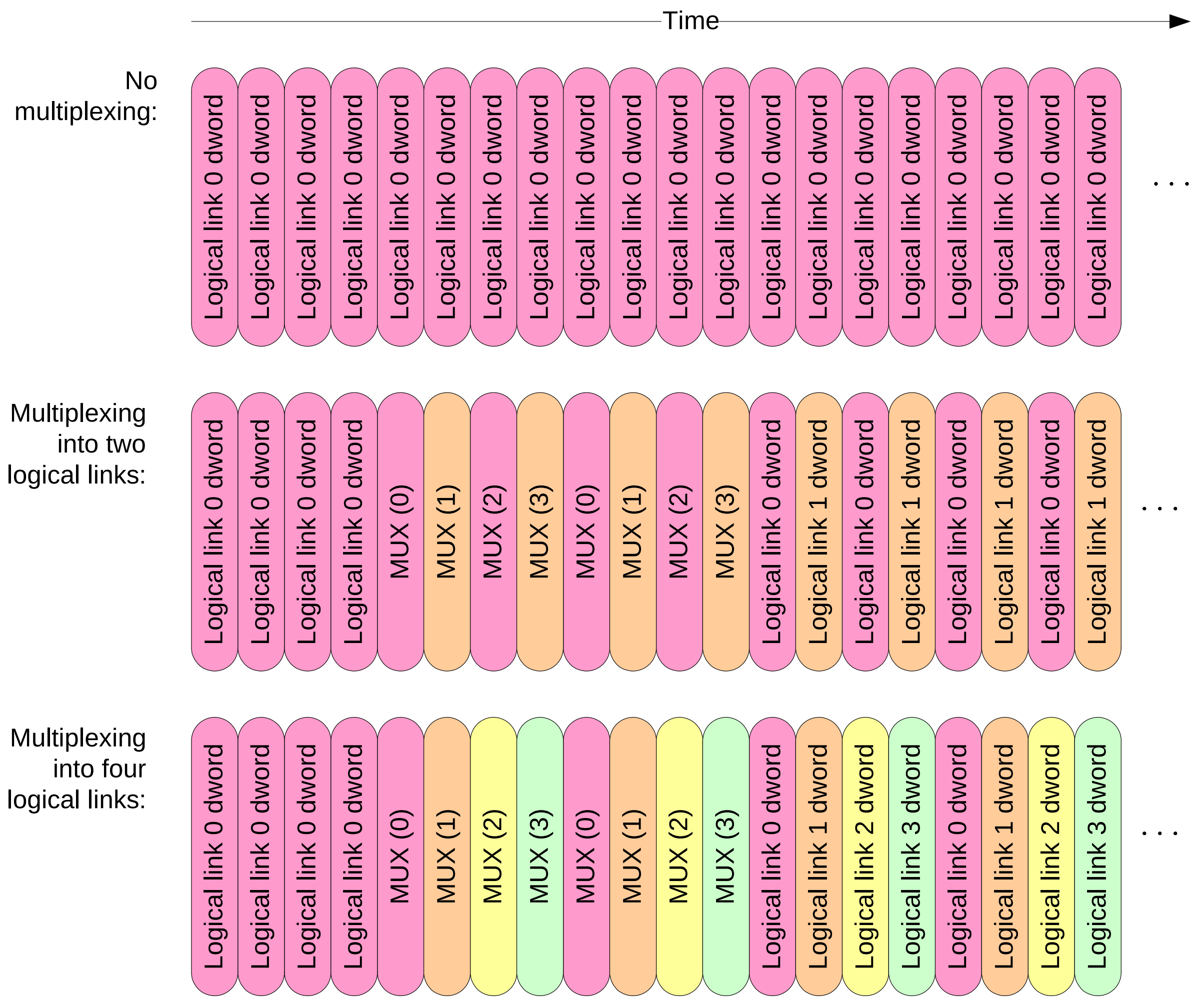


Figure 16 — Multiplexing

The first dword in a logical link may be any type of dword (e.g., ALIGN, NOTIFY, primitive, or data dword). Each logical phy shall honor the ALIGN insertion rate rules in 7.xx. The logical phys shall ignore MUX primitives.

The phy shall establish the incoming logical links based on the received MUX primitives (e.g., MUX (1) indicates the position of logical link 1). It shall receive 3 MUX primitives confirming each logical link before using the logical link.

The phy shall handle errors during the multiplexing sequence as follows:

a) If the phy receives a dword that is not a MUX primitive before receiving the MUX primitiive expected in that position, it shall discard the dword;
b) If the phy receives an invalid dword, it shall discard the dword;
c) If the phy receives a MUX primitive that does not match the MUX primitive expected in that position (e.g., it receives MUX(0) followed by MUX (2)), it shall shift the expected positions;
d) If the phy transmits MUX primitives for 1 ms without receiving MUX identifying the positions of each logical link, it shall restart the link reset sequence; and
e) If the phy finishes transmitting MUX primitives and starts transmitting non-MUX primitives, but does not stop receiving MUX primitives in all logical links for 1 ms, it shall restart the link reset sequence.

If the phy ever loses dword synchronization, it shall restart a link reset sequence rather than attempt to reestablish dword synchronization.

Once multiplexing sequence is complete, the phy shall not change multiplexing until a new link reset sequence. The phy shall not transmit MUX and shall ignore any incoming MUX primitives.

Editor's Note 10: Goals: Must tolerate single bit errors and seven bit burst errors, should tolerate multibit (possibly non-burst) errors. Errors could happen during the first MUX primitives, middle ones, or the last ones, or after multiplexing is established.

Editor's Note 11: Is there need to periodically validate the logical link numbers by sending MUX again (between connections)? If so, could define them as deletable (like ALIGN/NOTIFY) and allow them inside connections (but with no guarantee that there will be time slots to include them).

Changes to the port layer

None so far. It may need to be made clear that the port layer talks to logical phys, not physical phys.

Changes to the application layer

Define the SMP functions to enable multiplexing and discover if it is supported/being used.

10.4.3.5 DISCOVER function

The DISCOVER function returns the physical link configuration information for the specified phy. This SMP function provides information from the IDENTIFY address frame received by the phy and additional phy-specific information. This SMP function shall be implemented by all SMP target ports.

Table 10 defines the request format.

Table 10 — DISCOVER request

Byte\Bit	7	6	5	4	3	2	1	0
0	SMP FRAME TYPE (40h)							
1	FUNCTION (10h)							
2 … 8	Reserved							
9	PHY IDENTIFIER							
10 … 11	Reserved							
12 … 15	(MSB) CRC (LSB)							

The SMP FRAME TYPE field shall be set to 40h.

The FUNCTION field shall be set to 10h.

The PHY IDENTIFIER field specifies the phy (see 4.2.7) for the link configuration information being requested.

The CRC field is defined in 10.4.3.1.

Table 11 defines the response format.

Table 11 — DISCOVER response (part 1 of 2)

Byte\Bit	7	6	5	4	3	2	1	0
0	SMP FRAME TYPE (41h)							
1	FUNCTION (10h)							
2	FUNCTION RESULT							
3	RESPONSE LENGTH (0Dh)							
4 … 8	Reserved							
9	PHY IDENTIFIER							
10 … 11	Reserved							
12	Reserved	ATTACHED DEVICE TYPE			Reserved			
13	Reserved				NEGOTIATED PHYSICAL LINK RATE			
14	Reserved				ATTACHED SSP INITIATOR	ATTACHED STP INITIATOR	ATTACHED SMP INITIATOR	ATTACHED SATA HOST

Table 11 — DISCOVER response (part 2 of 2)

Byte\Bit	7	6	5	4	3	2	1	0
15	ATTACHED SATA PORT SELECTOR	Reserved			ATTACHED SSP TARGET	ATTACHED STP TARGET	ATTACHED SMP TARGET	ATTACHED SATA DEVICE
16	SAS ADDRESS							
23								
24	ATTACHED SAS ADDRESS							
31								
32	ATTACHED PHY IDENTIFIER							
33	Reserved							
39								
40	PROGRAMMED MINIMUM PHYSICAL LINK RATE				HARDWARE MINIMUM PHYSICAL LINK RATE			
41	PROGRAMMED MAXIMUM PHYSICAL LINK RATE				HARDWARE MAXIMUM PHYSICAL LINK RATE			
42	PHY CHANGE COUNT							
43	VIRTUAL PHY	Reserved			PARTIAL PATHWAY TIMEOUT VALUE			
44	Reserved				ROUTING ATTRIBUTE			
45	Reserved	CONNECTOR TYPE						
46	CONNECTOR ELEMENT INDEX							
47	CONNECTOR PHYSICAL LINK							
48	Reserved							
49								
50	Vendor specific							
51								
52	Reserved							
53								
54	Reserved				DESIRED LOGICAL LINK RATE			
55	Reserved		HARDWARE MAXIMUM LOGICAL LINKS		REQUESTED LOGICAL LINKS		ATTACHED REQUESTED LOGICAL LINKS	
~~52~~ 56	(MSB)	CRC						
~~55~~ 59								(LSB)

The SMP FRAME TYPE field shall be set to 41h.

The FUNCTION field shall be set to 10h.

The FUNCTION RESULT field is defined in 10.4.3.2.

The RESPONSE LENGTH field contains the number of dwords that follow, not including the CRC field (i.e., 13). A RESPONSE LENGTH field set to 00h indicates there are 12 additional dwords (i.e., 48 additional bytes) before the CRC field in the response frame.

The PHY IDENTIFIER field indicates the phy for which physical configuration link information is being returned.

The ATTACHED DEVICE TYPE field indicates the DEVICE TYPE value received during the link reset sequence and is defined in table 6.

Table 12 — ATTACHED DEVICE TYPE field

Code	Description
000b	No device attached
001b	End device
010b	Edge expander device
011b	Fanout expander device
All others	Reserved

The ATTACHED DEVICE TYPE field shall only be set to a value other than 000b after:

a) the identification sequence is complete if a SAS device or expander device is attached; or
b) the initial Register - Device to Host FIS has been received if a SATA phy is attached.

The NEGOTIATED PHYSICAL LINK RATE field is defined in table 13 and indicates the physical link rate negotiated during the link reset sequence. The negotiated physical link rate may be less than the programmed minimum physical link rate or greater than the programmed maximum physical link rate if the programmed physical link rates have been changed since the last link reset sequence.

Table 13 — NEGOTIATED PHYSICAL LINK RATE field

Code	Name	Description
0h	UNKNOWN	Phy is enabled; unknown physical link rate. [a]
1h	DISABLED	Phy is disabled.
2h	PHY_ RESET_ PROBLEM	Phy is enabled; the phy obtained dword synchronization for at least one physical link rate during the SAS speed negotiation sequence (see 6.7.4.2), but the SAS speed negotiation sequence failed (i.e., the last speed negotiation window, using a physical link rate expected to succeed, failed). These failures may be logged in the SMP REPORT PHY ERROR LOG function (see 10.4.3.6) and/or the Protocol-Specific Port log page (see 10.2.8.1).
3h	SPINUP_ HOLD	Phy is enabled; detected a SATA device and entered the SATA spinup hold state. The LINK RESET and HARD RESET operations in the SMP PHY CONTROL function (see) may be used to release the phy. This field shall be updated to this value at SATA spinup hold time (see 6.8.7 and 6.10)(i.e., after the COMSAS Detect Timeout timer expires during the SATA OOB sequence) if SATA spinup hold is supported.
4h	PORT_ SELECTOR	Phy is enabled; detected a SATA port selector. The physical link rate has not been negotiated since the last time the phy's SP state machine entered the SP0:OOB_COMINIT state. The SATA spinup hold state has not been entered since the last time the phy's SP state machine entered the SP0:OOB_COMINIT state. The value in this field may change to 3h, 8h, or 9h if attached to the active phy of the SATA port selector. Presence of a SATA port selector is indicated by the ATTACHED SATA PORT SELECTOR bit.
8h	G1	Phy is enabled; 1,5 Gbps physical link rate. This field shall be updated to this value after the speed negotiation sequence completes.
9h	G2	Phy is enabled; 3~~,0~~ Gbps physical link rate. This field shall be updated to this value after the speed negotiation sequence completes.
Ah	G3	Phy is enabled; 6 Gbps physical link rate. This field shall be updated to this value after the speed negotiation sequence completes.
All others	Reserved.	

[a] This code may be used by an application client in its local data structures to indicate an unknown negotiated physical link rate (e.g., before the discover process has queried the phy).

Table 14 describes the ATTACHED SATA PORT SELECTOR bit and the ATTACHED SATA DEVICE bit.

Table 14 — ATTACHED SATA PORT SELECTOR and ATTACHED SATA DEVICE bits

ATTACHED SATA PORT SELECTOR bit value [a] [b]	ATTACHED SATA DEVICE bit value [c] [d]	Description
0	0	Neither a SATA port selector nor a SATA device is attached and ready on the selected phy.
0	1	The attached phy is a SATA device phy. No SATA port selector is present (i.e., the SP state machine did not detect COMWAKE in response to the initial COMINIT, but sequenced through the normal (non-SATA port selector) SATA device OOB sequence).
1	0	The attached phy is a SATA port selector host phy, and either: a) the attached phy is the inactive host phy, or b) the attached phy is the active host phy and a SATA device is either not present or not ready behind the SATA port selector (i.e., the SP state machine detected COMWAKE while waiting for COMINIT).
1	1	The attached phy is a SATA port selector's active host phy and a SATA device is present behind the SATA port selector (i.e., the SP state machine detected COMWAKE while waiting for COMINIT, timed out waiting for COMSAS, and exchanged COMWAKE with an attached SATA device).

[a] The ATTACHED SATA PORT SELECTOR bit is invalid if the NEGOTIATED PHYSICAL LINK RATE field is set to UNKNOWN (i.e., 0h) or DISABLED (i.e., 1h).
[b] Whenever the ATTACHED SATA PORT SELECTOR bit changes, the phy shall generate a BROADCAST(CHANGE) notification.
[c] For the purposes of the ATTACHED SATA DEVICE bit, the SATA port selector is not considered a SATA device.
[d] The ATTACHED SATA DEVICE bit shall be updated at SATA spin-up hold time (see 6.8.7 and 6.10).

An ATTACHED SATA HOST bit set to one indicates a SATA host port is attached. An ATTACHED SATA HOST bit set to zero indicates a SATA host port is not attached.

NOTE 2 - Support for SATA hosts is outside the scope of this standard.

If a SAS phy reset sequence occurs (see 6.7.4)(i.e., one or more of the ATTACHED SSP INITIATOR PORT bit, ATTACHED STP INITIATOR PORT bit, the ATTACHED SMP INITIATOR PORT bit, the ATTACHED SSP TARGET PORT bit, the ATTACHED STP TARGET PORT bit, and/or the ATTACHED SMP TARGET PORT bit is set to one), then the ATTACHED SATA PORT SELECTOR bit, the ATTACHED SATA DEVICE bit, and the ATTACHED SATA HOST bit shall each be set to zero.

The ATTACHED SSP INITIATOR PORT bit indicates the value of the SSP INITIATOR PORT field received in the IDENTIFY address frame (see 7.8.2) during the identification sequence.

The ATTACHED STP INITIATOR PORT bit indicates the value of the STP INITIATOR PORT field received in the IDENTIFY address frame (see 7.8.2) during the identification sequence.

The ATTACHED SMP INITIATOR PORT bit indicates the value of the SMP INITIATOR PORT field received in the IDENTIFY address frame (see 7.8.2) during the identification sequence.

The ATTACHED SSP TARGET PORT bit indicates the value of the SSP TARGET PORT field received in the IDENTIFY address frame (see 7.8.2) during the identification sequence.

The ATTACHED STP TARGET PORT bit indicates the value of the STP TARGET PORT field received in the IDENTIFY address frame (see 7.8.2) during the identification sequence.

The ATTACHED SMP TARGET PORT bit indicates the value of the SMP TARGET PORT field received in the IDENTIFY address frame (see 7.8.2) during the identification sequence.

The ATTACHED SSP INITIATOR PORT bit, ATTACHED STP INITIATOR PORT bit, ATTACHED SMP INITIATOR PORT bit, ATTACHED SSP TARGET PORT bit, ATTACHED STP TARGET PORT bit, and ATTACHED SMP TARGET PORT bit shall be updated at the end of the identification sequence.

If a SATA phy reset sequence occurs (see 6.7.3)(i.e., the ATTACHED SATA PORT SELECTOR bit is set to one, the ATTACHED SATA DEVICE bit is set to one, or the ATTACHED SATA HOST bit is set to one), then the ATTACHED SSP INITIATOR PORT bit, ATTACHED STP INITIATOR PORT bit, ATTACHED SMP INITIATOR PORT bit, ATTACHED SSP TARGET PORT bit, ATTACHED STP TARGET PORT bit, and ATTACHED SMP TARGET PORT bit shall each be set to zero.

The SAS ADDRESS field contains the value of the SAS ADDRESS field transmitted in the IDENTIFY address frame during the identification sequence. If the phy is an expander phy, the SAS ADDRESS field contains the SAS address of the expander device (see 4.2.4). If the phy is a SAS phy, the SAS ADDRESS field contains the SAS address of the SAS port (see 4.2.6).

The ATTACHED SAS ADDRESS field contains the value of the SAS ADDRESS field received in the IDENTIFY address frame during the identification sequence. If the attached port is an expander port, the ATTACHED SAS ADDRESS field contains the SAS address of the attached expander device (see 4.2.4). If the attached port is a SAS port, the ATTACHED SAS ADDRESS field contains SAS address of the attached SAS port (see 4.2.6). If the attached port is a SATA device port, the ATTACHED SAS ADDRESS field contains the SAS address of the STP/SATA bridge (see 4.6.2).

The ATTACHED SAS ADDRESS field shall be updated:

a) after the identification sequence completes, if a SAS phy or expander phy is attached; or
b) after the COMSAS Detect Timeout timer expires (see 6.8.3.9), if a SATA phy is attached.

An STP initiator port should not make a connection request to the attached SAS address until the ATTACHED DEVICE TYPE field is set to a value other than 000b.

The ATTACHED PHY IDENTIFIER field contains a phy identifier for the attached phy:

a) If the attached phy is a SAS phy or an expander phy, the ATTACHED PHY IDENTIFIER field contains the value of the PHY IDENTIFIER field received in the IDENTIFY address frame during the identification sequence:
    A) If the attached phy is a SAS phy, the ATTACHED PHY IDENTIFIER field contains the phy identifier of the attached SAS phy in the attached SAS device;
    B) If the attached phy is an expander phy, the ATTACHED PHY IDENTIFIER field contains the phy identifier (see 4.2.7) of the attached expander phy in the attached expander device; and
b) If the attached phy is a SATA device phy, the ATTACHED PHY IDENTIFIER field contains 00h;
c) If the attached phy is a SATA port selector phy and the expander device is able to determine the port of the SATA port selector to which it is attached, the ATTACHED PHY IDENTIFIER field contains 00h or 01h; and
d) If the attached phy is a SATA port selector phy and the expander device is not able to determine the port of the SATA port selector to which it is attached, the ATTACHED PHY IDENTIFIER field contains 00h.

The ATTACHED PHY IDENTIFIER field shall be updated:

a) after the identification sequence completes, if a SAS phy or expander phy is attached; or
b) after the COMSAS Detect Timeout timer expires (see 6.8.3.9), if a SATA phy is attached.

The PROGRAMMED MINIMUM PHYSICAL LINK RATE field indicates the minimum physical link rate set by the PHY CONTROL function (see). The values are defined in table 15. The default value shall be the value of the HARDWARE MINIMUM PHYSICAL LINK RATE field.

The HARDWARE MINIMUM PHYSICAL LINK RATE field indicates the minimum physical link rate supported by the phy. The values are defined in table 16.

The PROGRAMMED MAXIMUM PHYSICAL LINK RATE field indicates the maximum physical link rate set by the PHY CONTROL function (see). The values are defined in table 15. The default value shall be the value of the HARDWARE MAXIMUM PHYSICAL LINK RATE field.

Table 15 — PROGRAMMED MINIMUM PHYSICAL LINK RATE and PROGRAMMED MAXIMUM PHYSICAL LINK rate fields

Code	Description
0h	Not programmable
8h	1,5 Gbps
9h	3,0 Gbps
Ah	6 Gbps
All others	Reserved

The HARDWARE MAXIMUM PHYSICAL LINK RATE field indicates the maximum physical link rate supported by the phy. The values are defined in table 16.

Table 16 — HARDWARE MINIMUM PHYSICAL LINK RATE and HARDWARE MAXIMUM PHYSICAL LINK RATE fields

Code	Description
8h	1,5 Gbps
9h	3,0 Gbps
Ah	6 Gbps
All others	Reserved

The PHY CHANGE COUNT field counts the number of BROADCAST (CHANGE)s originated by an expander phy. Expander devices shall support this field. Other device types shall not support this field. This field shall be set to zero at power on. The expander device shall increment this field at least once when it transmits a BROADCAST (CHANGE) for any reason described in 7.11 originating from the expander phy other than forwarding a BROADCAST (CHANGE).

The expander device is not required to increment the PHY CHANGE COUNT field again unless a DISCOVER response is transmitted. This field shall not be incremented when forwarding a BROADCAST (CHANGE) from another expander device.The PHY CHANGE COUNT field shall wrap to zero after the maximum value (i.e., FFh) has been reached.

NOTE 3 - Application clients that use the PHY CHANGE COUNT field should read it often enough to ensure that it does not increment a multiple of 256 times between reading the field.

A VIRTUAL PHY bit set to one indicates the phy is part of an internal port and the attached device is contained within the expander device. A VIRTUAL PHY bit set to zero indicates the phy is a physical phy and the attached device is not contained within the expander device.

The PARTIAL PATHWAY TIMEOUT VALUE field indicates the partial pathway timeout value in microseconds (see 7.12.4.5).

NOTE 4 - The recommended default value for PARTIAL PATHWAY TIMEOUT VALUE is 7 µs. The partial pathway timeout value may be set by the PHY CONTROL function (see).

The ROUTING ATTRIBUTE field indicates the routing attribute supported by the phy (see 4.6.7.1) and is defined in table 17.

Table 17 — ROUTING ATTRIBUTE field

Code	Name	Description
0h	Direct routing attribute	Direct routing method for attached end devices. Attached expander devices are not supported on this phy.
1h	Subtractive routing attribute	Either: a) subtractive routing method for attached expander devices; or b) direct routing method for attached end devices.
2h	Table routing attribute	Either: a) table routing method for attached expander devices; or b) direct routing method for attached end devices.
All others	Reserved	

The ROUTING ATTRIBUTE field shall not change based on the attached device type.

The CONNECTOR TYPE field indicates the type of connector used to access the phy, as reported by the enclosure services process for the enclosure (see the SAS Connector element in SES-2). A CONNECTOR TYPE field set to 00h indicates no connector information is available and that the CONNECTOR ELEMENT INDEX field and the CONNECTOR PHYSICAL LINK fields are invalid and shall be ignored.

The CONNECTOR ELEMENT INDEX indicates the element index of the SAS Connector element representing the connector used to access the phy, as reported by the enclosure services process for the enclosure (see the SAS Connector element in SES-2).

The CONNECTOR PHYSICAL LINK field indicates the physical link in the connector used to access the phy, as reported by the enclosure services process for the enclosure (see the SAS Connector element in SES-2).

The DESIRED LOGICAL LINK RATE field indicates the value of the DESIRED LOGICAL LINK RATE field set by the PHY CONTROL function and is defined in table 7 (see 7.xx). The default value shall be the minimum rate that supports the bandwidth indicated by the REQUESTED BANDWIDTH field (e.g., if the requested bandwidth is 320 MBps, this field shall be set to 6 Gbps).

Table 18 — DESIRED LOGICAL LINK RATE field

Code	Description
8h	1,5 Gbps
9h	3~~,0~~ Gbps
Ah	6 Gbps
All others	Reserved

Editor's Note 12: The default value is important. If the default is the maximum physical link rate, then the phys power up without multiplexing and a management application client needs to make an effort to turn it on. If the default is 1.5 Gbps, then phys will power up with multiplexing and a management application client will have to make an effort to turn it off. The default proposed is the minimum rate that the phy wants to meet its bandwidth needs (e.g. 6 Gbps for 320 MBps).

The HARDWARE MAXIMUM LOGICAL LINKS field indicates the maximum value supported by the phy for the REQUESTED LOGICAL LINKS field in the IDENTIFY address frame and is defined in table 7 (see 7.xx). This value is not adjusted based on the current physical link rate.

The REQUESTED LOGICAL LINKS field indicates the value of the REQUESTED LOGICAL LINKS field transmitted during the link reset sequence and is defined in table 7 (see 7.xx).

The ATTACHED REQUESTED LOGICAL LINKS field indicates the value of the REQUESTED LOGICAL LINKS field received during the link reset sequence and is defined in table 7 (see 7.xx).

The CRC field is defined in 10.4.3.2.

10.4.3.10 PHY CONTROL function

The PHY CONTROL function requests actions by the specified phy. This SMP function may be implemented by any SMP target port.

Table 19 defines the request format.

Table 19 — PHY CONTROL request

Byte\Bit	7	6	5	4	3	2	1	0
0	SMP FRAME TYPE (40h)							
1	FUNCTION (91h)							
2	Reserved							
3	REQUEST LENGTH (09h)							
4 … 8	Reserved							
9	PHY IDENTIFIER							
10	PHY OPERATION							
11	Reserved							UPDATE PARTIAL PATHWAY TIMEOUT VALUE
12 … 31	Reserved							
32	PROGRAMMED MINIMUM PHYSICAL LINK RATE				Reserved			
33	PROGRAMMED MAXIMUM PHYSICAL LINK RATE				Reserved			
34 … 35	Reserved							
36	Reserved				PARTIAL PATHWAY TIMEOUT VALUE			
37	Reserved				DESIRED LOGICAL LINK RATE			
38 … 39	Reserved							
40 … 43	(MSB) CRC (LSB)							

The SMP FRAME TYPE field shall be set to 40h.

The FUNCTION field shall be set to 91h.

The REQUEST LENGTH field contains the number of dwords that follow, not including the CRC field (i.e., 9). A REQUEST LENGTH field set to 00h indicates there are 9 additional dwords (i.e., 36 additional bytes) before the CRC field in the request frame.

The PHY IDENTIFIER field specifies the phy (see 4.2.7) to which the PHY CONTROL request applies.

Table 20 defines the PHY OPERATION field.

Table 20 — PHY OPERATION field (part 1 of 2)

Code	Operation	Description
00h	NOP	No operation.
01h	LINK RESET	If the specified phy is not a virtual phy, perform a link reset sequence (see 4.4) on the specified phy and enable the specified phy. If the specified phy is a virtual phy, perform an internal reset and enable the specified phy. See 7.11 for BROADCAST (CHANGE) requirements related to this phy operation in an expander device. Any affiliation (see 7.17.5) shall continue to be present. The phy shall bypass the SATA spinup hold state, if implemented (see 6.8.3.9). The SMP response shall be returned without waiting for the link reset to complete.
02h	HARD RESET	If the specified phy is not a virtual phy, perform a link reset sequence (see 4.4) on the specified phy and enable the specified phy. If the attached phy is a SAS phy or an expander phy, the link reset sequence shall include a hard reset sequence (see 4.4.2). If the attached phy is a SATA phy, the phy shall bypass the SATA spinup hold state. See 7.11 for BROADCAST (CHANGE) requirements related to this phy operation in an expander device. If the specified phy is a virtual phy, perform an internal reset and enable the specified phy. Any affiliation (see 7.17.5) shall be cleared. The SMP response shall be returned without waiting for the hard reset to complete.
03h	DISABLE	Disable the specified phy (i.e., stop transmitting valid dwords and receiving dwords on the specified phy). The LINK RESET and HARD RESET operations may be used to enable the phy. See 7.11 for BROADCAST (CHANGE) requirements related to this phy operation in an expander device.
04h	Reserved	
05h	CLEAR ERROR LOG	Clear the error log counters (see 10.4.3.6) for the specified phy.

Table 20 — PHY OPERATION field (part 2 of 2)

Code	Operation	Description
06h	CLEAR AFFILIATION	Clear an affiliation (see 7.17.5) from the STP initiator port with the same SAS address as the SMP initiator port that opened this SMP connection. If there is no such affiliation, the SMP target port shall return a function result of SMP FUNCTION FAILED in the response frame.
07h	TRANSMIT SATA PORT SELECTION SIGNAL	This function shall only be supported by phys in an expander device. If the expander phy incorporates an STP/SATA bridge and supports SATA port selectors, the phy shall transmit the SATA port selection signal (see 6.6) which causes the SATA port selector to select the attached phy as the active host phy and make its other host phy inactive. See 7.11 for BROADCAST (CHANGE) requirements related to this phy operation in an expander device. Any affiliation (see 7.17.5) shall be cleared. If the expander phy does not support SATA port selectors, then the SMP target port shall return a function result of PHY DOES NOT SUPPORT SATA. If the expander phy supports SATA port selectors but is attached to a SAS phy or an expander phy, the SMP target port shall return a function result of SMP FUNCTION FAILED.
All others	Reserved	

If the PHY IDENTIFIER field specifies the phy which is being used for the SMP connection and a phy operation of LINK RESET, HARD RESET, or DISABLE is requested, the SMP target port shall not perform the requested operation and shall return a function result of SMP FUNCTION FAILED in the response frame.

An UPDATE PARTIAL PATHWAY TIMEOUT VALUE bit set to one specifies that the PARTIAL PATHWAY TIMEOUT VALUE field shall be honored. An UPDATE PARTIAL PATHWAY TIMEOUT VALUE bit set to zero specifies that the PARTIAL PATHWAY TIMEOUT VALUE field shall be ignored.

The PROGRAMMED MINIMUM PHYSICAL LINK RATE field specifies the minimum physical link rate the phy shall support during a link reset sequence (see 4.4.1). Table 21 defines the values for this field. If this field is changed along with a phy operation of LINK RESET or HARD RESET, that phy operation shall utilize the new value for this field.

The PROGRAMMED MAXIMUM PHYSICAL LINK RATE field specifies the maximum physical link rates the phy shall support during a link reset sequence (see 4.4.1).Table 21 defines the values for this field. If this field is changed along with a phy operation of LINK RESET or HARD RESET, that phy operation shall utilize the new value for this field.

Table 21 — PROGRAMMED MINIMUM PHYSICAL LINK RATE and PROGRAMMED MAXIMUM PHYSICAL LINK RATE fields

Code	Description
0h	Do not change current value
8h	1,5 Gbps
9h	3~~,0~~ Gbps
Ah	6 Gbps
All others	Reserved

If the PROGRAMMED MINIMUM PHYSICAL LINK RATE field or the PROGRAMMED MAXIMUM PHYSICAL LINK RATE field is set to an unsupported or reserved value, or the PROGRAMMED MINIMUM PHYSICAL LINK RATE field and PROGRAMMED MAXIMUM PHYSICAL LINK RATE field are set to an invalid combination of values (e.g., the minimum is greater than the maximum), the SMP target port shall not change either of their values and may return a function result of SMP FUNCTION FAILED in the response frame. If it returns a function result of SMP FUNCTION FAILED, it shall not perform the requested phy operation.

The PARTIAL PATHWAY TIMEOUT VALUE field specifies the amount of time in microseconds the expander phy shall wait after receiving an Arbitrating (Blocked On Partial) confirmation from the ECM before requesting that the ECM resolve pathway blockage (see 7.12.4.6). A PARTIAL PATHWAY TIMEOUT VALUE field value of zero (i.e., 0 µs) specifies that partial pathway resolution shall be requested by the expander phy immediately upon reception of an Arbitrating (Blocked On Partial) confirmation from the ECM. The PARTIAL PATHWAY TIMEOUT VALUE field is only honored when the UPDATE PARTIAL PATHWAY TIMEOUT VALUE bit is set to one.

The DESIRED LOGICAL LINK RATE field specifies the logical link rate the phy should attempt to enable via multiplexing and is defined in table 22. If this field is changed along with a phy operation of LINK RESET or HARD RESET, that phy operation shall utilize the new value for this field.

Table 22 — DESIRED LOGICAL LINK RATE field

Code	**Description**
0h	Do not change current value
8h	1,5 Gbps
9h	3 Gbps
Ah	6 Gbps
All others	Reserved

The CRC field is defined in 10.4.3.1.

Table 23 defines the response format.

Table 23 — PHY CONTROL response

Byte\Bit	7	6	5	4	3	2	1	0
0	SMP FRAME TYPE (41h)							
1	FUNCTION (91h)							
2	FUNCTION RESULT							
3	Reserved							
4	(MSB) CRC							
7								(LSB)

The SMP FRAME TYPE field shall be set to 41h.

The FUNCTION field shall be set to 91h.

The FUNCTION RESULT field is defined in 10.4.3.2.

The CRC field is defined in 10.4.3.2.

Changes to the annexes

Define the MUX primitive encodings.

Annex J Primitive encoding

The MUX primitive encodings were selected to avoid having as many duplicate characters as possible (to reduce EMI, since these primitives are transmitted back-to-back). There are two overlaps in this set (D16.7 and D24.0), which is the best available in the unused values.

...

Table 0.1 — Primitives with Hamming distance of 8 (part 1 of 3)

1st	2nd	3rd	4th	Assignment
K28.5	D01.3	D01.3	D01.3	ALIGN (2)
K28.5	D01.4	D01.4	D01.4	ACK
K28.5	D01.4	D02.0	D31.4	RRDY (RESERVED 0)
K28.5	D01.4	D04.7	D24.0	NAK (RESERVED 1)
K28.5	D01.4	D07.3	D30.0	CREDIT_BLOCKED
K28.5	D01.4	D16.7	D07.3	NAK (RESERVED 2)
K28.5	D01.4	D24.0	D16.7	RRDY (NORMAL)
K28.5	D01.4	D27.4	D04.7	NAK (CRC ERROR)
K28.5	D01.4	D30.0	D02.0	RRDY (RESERVED 1)
K28.5	D01.4	D31.4	D29.7	NAK (RESERVED 0)
K28.5	D02.0	D01.4	D29.7	ERROR
K28.5	D02.0	D02.0	D02.0	HARD_RESET
K28.5	D02.0	D04.7	D01.4	CLOSE (RESERVED 1)
K28.5	D02.0	D07.3	D04.7	CLOSE (CLEAR AFFILIATION)
K28.5	D02.0	D16.7	D31.4	MUX (0)
K28.5	D02.0	D24.0	D07.3	BREAK
K28.5	D02.0	D29.7	D16.7	
K28.5	D02.0	D30.0	D27.4	CLOSE (NORMAL)
K28.5	D02.0	D31.4	D30.0	CLOSE (RESERVED 0)
K28.5	D04.7	D01.4	D24.0	BROADCAST (RESERVED 1)
K28.5	D04.7	D02.0	D01.4	BROADCAST (CHANGE)
K28.5	D04.7	D04.7	D04.7	BROADCAST (RESERVED 2)
K28.5	D04.7	D07.3	D29.7	BROADCAST (SES)
K28.5	D04.7	D16.7	D02.0	BROADCAST (RESERVED 3)
K28.5	D04.7	D24.0	D31.4	BROADCAST (RESERVED CHANGE 0)
K28.5	D04.7	D27.4	D07.3	BROADCAST (RESERVED CHANGE 1)
K28.5	D04.7	D29.7	D30.0	BROADCAST (RESERVED 4)
K28.5	D04.7	D31.4	D27.4	
K28.5	D07.0	D07.0	D07.0	ALIGN (1)
K28.5	D07.3	D01.4	D31.4	
K28.5	D07.3	D02.0	D04.7	
K28.5	D07.3	D04.7	D30.0	MUX (1)
K28.5	D07.3	D07.3	D07.3	

Table 0.1 — Primitives with Hamming distance of 8 (part 2 of 3)

1st	2nd	3rd	4th	Assignment
K28.5	D07.3	D24.0	D29.7	
K28.5	D07.3	D27.4	D16.7	
K28.5	D07.3	D29.7	D27.4	
K28.5	D07.3	D30.0	D24.0	
K28.5	D07.3	D31.4	D02.0	
K28.5	D10.2	D10.2	D27.3	ALIGN (0)
K28.5	D16.7	D01.4	D02.0	
K28.5	D16.7	D02.0	D07.3	
K28.5	D16.7	D04.7	D31.4	
K28.5	D16.7	D16.7	D16.7	OPEN_ACCEPT
K28.5	D16.7	D24.0	D27.4	MUX (2)
K28.5	D16.7	D27.4	D30.0	
K28.5	D16.7	D29.7	D24.0	
K28.5	D16.7	D30.0	D04.7	
K28.5	D16.7	D31.4	D01.4	
K28.5	D24.0	D01.4	D16.7	MUX (3)
K28.5	D24.0	D02.0	D29.7	
K28.5	D24.0	D04.7	D07.3	SOF
K28.5	D24.0	D07.3	D31.4	EOAF
K28.5	D24.0	D16.7	D27.4	EOF
K28.5	D24.0	D24.0	D24.0	
K28.5	D24.0	D27.4	D02.0	
K28.5	D24.0	D29.7	D04.7	
K28.5	D24.0	D30.0	D01.4	SOAF
K28.5	D27.3	D27.3	D27.3	ALIGN (3)
K28.5	D27.4	D01.4	D07.3	AIP (RESERVED WAITING ON PARTIAL)
K28.5	D27.4	D04.7	D02.0	
K28.5	D27.4	D07.3	D24.0	AIP (WAITING ON CONNECTION)
K28.5	D27.4	D16.7	D30.0	AIP (RESERVED 1)
K28.5	D27.4	D24.0	D04.7	AIP (WAITING ON PARTIAL)
K28.5	D27.4	D27.4	D27.4	AIP (NORMAL)
K28.5	D27.4	D29.7	D01.4	AIP (RESERVED 2)
K28.5	D27.4	D30.0	D29.7	AIP (WAITING ON DEVICE)
K28.5	D27.4	D31.4	D16.7	AIP (RESERVED 0)
K28.5	D29.7	D02.0	D30.0	OPEN_REJECT (RESERVED CONTINUE 0)
K28.5	D29.7	D04.7	D27.4	OPEN_REJECT (RESERVED STOP 1)
K28.5	D29.7	D07.3	D16.7	OPEN_REJECT (RESERVED INITIALIZE 1)
K28.5	D29.7	D16.7	D04.7	OPEN_REJECT (PATHWAY BLOCKED)

Table 0.1 — Primitives with Hamming distance of 8 (part 3 of 3)

1st	2nd	3rd	4th	Assignment
K28.5	D29.7	D24.0	D01.4	OPEN_REJECT (RESERVED CONTINUE 1)
K28.5	D29.7	D27.4	D24.0	OPEN_REJECT (RETRY)
K28.5	D29.7	D29.7	D29.7	OPEN_REJECT (NO DESTINATION)
K28.5	D29.7	D30.0	D31.4	OPEN_REJECT (RESERVED INITIALIZE 0)
K28.5	D29.7	D31.4	D07.3	OPEN_REJECT (RESERVED STOP 0)
K28.5	D30.0	D01.4	D04.7	DONE (ACK/NAK TIMEOUT)
K28.5	D30.0	D02.0	D16.7	
K28.5	D30.0	D07.3	D27.4	DONE (CREDIT TIMEOUT)
K28.5	D30.0	D16.7	D01.4	DONE (RESERVED 0)
K28.5	D30.0	D24.0	D02.0	
K28.5	D30.0	D27.4	D29.7	DONE (RESERVED TIMEOUT 0)
K28.5	D30.0	D29.7	D31.4	DONE (RESERVED 1)
K28.5	D30.0	D30.0	D30.0	DONE (NORMAL)
K28.5	D30.0	D31.4	D24.0	DONE (RESERVED TIMEOUT 1)
K28.5	D31.3	D01.3	D07.0	NOTIFY (RESERVED 1)
K28.5	D31.3	D07.0	D01.3	NOTIFY (POWER FAILURE EXPECTED)
K28.5	D31.3	D10.2	D10.2	NOTIFY (RESERVED 2)
K28.5	D31.3	D31.3	D31.3	NOTIFY (ENABLE SPINUP)
K28.5	D31.4	D01.4	D30.0	OPEN_REJECT (RESERVED ABANDON 3)
K28.5	D31.4	D02.0	D27.4	OPEN_REJECT (RESERVED ABANDON 0)
K28.5	D31.4	D04.7	D29.7	OPEN_REJECT (CONNECTION RATE NOT SUPPORTED)
K28.5	D31.4	D07.3	D02.0	OPEN_REJECT (RESERVED ABANDON 2)
K28.5	D31.4	D16.7	D24.0	OPEN_REJECT (WRONG DESTINATION)
K28.5	D31.4	D27.4	D01.4	OPEN_REJECT (STP RESOURCES BUSY)
K28.5	D31.4	D29.7	D07.3	OPEN_REJECT (PROTOCOL NOT SUPPORTED)
K28.5	D31.4	D30.0	D16.7	OPEN_REJECT (RESERVED ABANDON 1)
K28.5	D31.4	D31.4	D31.4	OPEN_REJECT (BAD DESTINATION)